Shortest Path Planning and Robust Control of Two-wheeled Mobile Robot
이륜구동로봇의 최단거리계획과 강인제어

김학경
H. K. Kim

Key Words : Path Planning(경로계획), Sliding Mode Control(슬라이딩 모드 제어), Two-wheeled Mobile Robot (이륜구동로봇)

Abstract : 본 논문은 Dijkstra 알고리즘에 기초한 최단거리 경로계획을 하며 이 경로를 추적하기 위한 슬라이딩 모드 제어를 제시한다. 슬라이딩 모드 제어기는 동적매개변수 불확실성과 입력외란이 존재 시에도 강인 점근적으로 계획된 경로를 추적하도록 한다. 더불어 작업장 내의 이동로봇의 위치를 USB 카메라에 의해 감지하며, Pin-hole 카메라모델로 하여 카메라에 의해 관측되는 작업장 내의 이륜구동로봇의 위치좌표를 결정하였으며, 이 위치를 정확히 감지하기 위해 Tsai법을 사용하여 카메라 보정한다. 시뮬레이션 결과는 슬라이딩 모드 제어기의 성능을 검증하기 위해 보였다.

1. Introduction

A two-wheeled mobile robot has been used in several industrial works such as welding, painting, etc. There can be obstacles in their work spaces. The robot needs a path planning to avoid these obstacles.

There were the several literatures about the path planning as follows. S. Sundar, etc. proposed the shortest path from an initial point to the target using HJB path planning method[5]. A. Hoover, etc. concentrated on path characteristics, such as smoothness and continuousness using polynomial approach[3].

For the mobile robot to track the path generated by the path planning method, a control method for mobile robot is needed. The several researchers proposed the control methods as follows. Fierro proposed transformation of dynamic model into kinematic model and computed torque control law using backstepping approach based on Lyapunov function[4]. D. K. Chwa et al. proposed a sliding mode controller for trajectory tracking of a nonholonomic wheeled mobile robots under the presence of the external disturbances represented as two-dimensional polar coordinates[1]. The mobile robot dynamic parameters are assumed to be totally known in the two above controllers. T. Fukao proposed the integration of a kinematic adaptive controller and a torque controller of a nonholonomic mobile robot with the dynamic parametric uncertainties[2]. These controllers can ensure robustness only with respect to input disturbances but the parameter variations cannot be dealt with.

In this paper, firstly, the shortest path is planned based on Dijkstra's algorithm, and then the path is smoothened using cubic spline algorithm. Secondly, even in the presence of uncertainties of both parameter variations and input disturbances, the sliding mode controller (SMC) for a mobile robot to track the planned path is proposed . The proposed controller is based on two nonlinear sliding surfaces ensuring the tracking of the three output variables exploiting the nonholonomic constraint. This implies the vanishing of the orientation error on the second sliding surface. Thirdly, a USB camera is calibrated using Tsai's method to detect

접수일 : 2006년 11월 20일, 채택확정 : 2006년 11월 28일
김학경(책임저자) : 부경대학교 기계공학부
E-mail : hakyeong@hanmai.net Tel. 051-620-1606

the mobile robot localization exactly. Finally, the simulation results show the effectiveness of the system model, the proposed controller and the camera calibration method.

2. Path Planning for Mobile Robot

The problem to avoid these obstacles is to find a collision free path for a mobile robot with two degree of freedom between two given points. The general idea for the path planning problems can be stated with the following assumptions:

(1) The obstacles are considered to be 2D convex or concave polygonal with unknown vertices, and a mobile robot is a 2D convex polygonal.

(2) The obstacles are grown by the size of the mobile robot, thereby reducing the analysis of the robot's motion from a moving area to a single moving point.

In this paper, the simplest map called the *visibility graph* is considered.

2.1 Visibility graph

A standard visibility graph is defined in a two-dimensional polygonal configuration space. A visibility graph is an undirected graph $G=(V,E)$ where V is the set of vertices ν_i of the grown obstacles O_i plus the START and GOAL points, and E is a set of edges e_{ij}, (i,j) are the index of the obstacles which are straight-line segments that connect two line-of-sight vertices ν_i and ν_j. Rather, they consist of all polygonal obstacle boundary edges and edges between any two vertices in V that lies entirely in free space except for its endpoints.

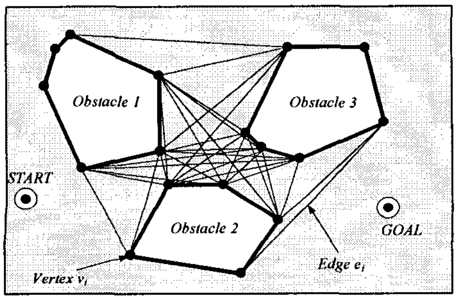


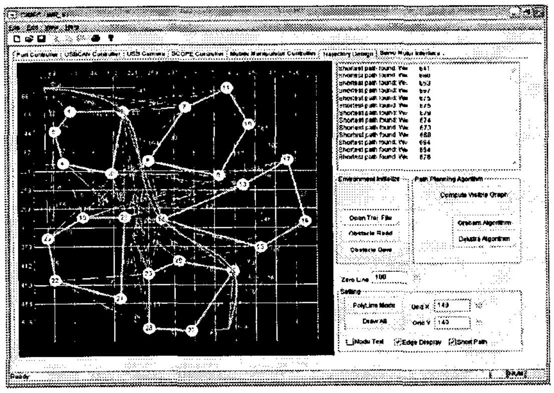
Fig. 1 Visibility graph

2.2 Shortest Path Finding - Dijkstra's Algorithm

The pseudocode of the Dijkstra's algorithm is as follows:

```
//****************************************
function Dijkstra($G,w,s$)
for each vertex $\nu$ in $V[G]$]//Initialization
  do $d[\nu]$ := infinity
      previous[$\nu$] := undefined
  $d[s]$ := 0
  $S$ := empty set
  $Q$ := set of all vertices
  while  is not an empty set
    do $u$ := Extract-Min($Q$)
        $S$ :=$S$ union $\{u\}$
  for each edge $(u,\nu)$ outgoing from u
    do if $d[\nu]>d(u)+w(u,\nu)$//Relax $(u,\nu)$
        then $d[\nu]]$ :=$d(u)+w(u,\nu)$
                previous[$\nu$] := $u$
                $Q$ := Update($Q$)
//****************************************
```

The shortest path needs to be smooth, and cubic spline algorithm can be applied totally. The result trajectory of cubic spline is shown in Fig. 2.


Shortest path found: We 641
Shortest path found: We 660
Shortest path found: We 653
Shortest path found: We 667
Shortest path found: We 675
Shortest path found: We 675
Shortest path found: We 679
Shortest path found: We 674
Shortest path found: We 673
Shortest path found: We 688
Shortest path found: We 664
Shortest path found: We 654
Shortest path found: We 678

Zero Line 100

Grid X 140

Grid Y 140

☐ Node Text ☑ Edge Display ☑ Short Path

Fig. 2 Dijkstra and cubic spline implementation

3. Robust SMC for Mobile Robot

3.1 Dynamic model of the WMR

The model of the two-wheeled mobile robot is shown in Fig. 3.

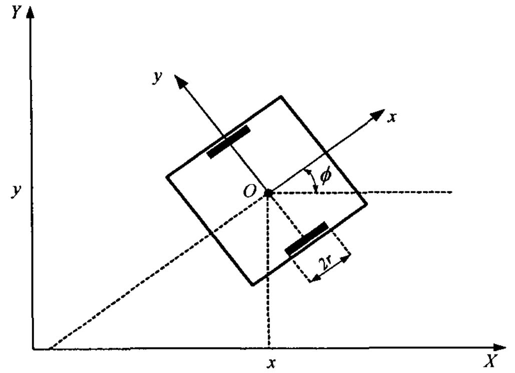


Fig. 3 Model of the two-wheeled mobile robot

Lagrange equation of motion for the non-holonomic mobile platform system are given by:

$$\ddot{z}=f(z)+g(z)(u+d) \tag{1}$$

where

$$f(z)=b_1\begin{bmatrix}-\lambda\sin\Phi\\ \lambda\cos\phi\\ 0\end{bmatrix},\quad g(z)=\begin{bmatrix}b_1\cos\phi & 0\\ b_1\sin\phi & 0\\ 0 & b_2\end{bmatrix},$$

$$u=\begin{bmatrix}u_1\\ u_2\end{bmatrix}=\begin{bmatrix}\tau_1+\tau_2\\ \tau_1-\tau_2\end{bmatrix},\quad d=\begin{bmatrix}d_1\\ d_2\end{bmatrix}=\begin{bmatrix}\tau_{1d}+\tau_{d2}\\ \tau_{d1}-\tau_{d2}\end{bmatrix}$$

$$b_1=\frac{1}{r\left(m+\frac{2I_w}{r^2}\right)},\ b_2=\frac{b}{r\left(I+\frac{I_w}{2c^2}\right)},$$

$$\lambda=\left(m+\frac{2I_w}{r^2}\right)(\dot{x}\cos\phi+\dot{y}\sin\phi)\phi+\dot{m}_c\,d\,\dot{\omega}$$

where the posture of the mobile robot be $z=[x\ y\ \phi]^T$, (x,y) is center position and ϕ is rotation angle of mobile platform, m, I_w are the mass of the WMR and moment of inertia of wheel, r is wheel radius, b is distance between center position and symmetry axis of mobile platform, τ_1,τ_2 are torques of right and left wheels, λ is Lagrange multiplier and $\tau_{d1,}\tau_{d2}$ are is unknown bounded input disturbances.The assumption that the signs of $b_1$and b_2 are known is practical since $b_1$and b_2 are represented as combinations of the robot's mass, moment of inertia, wheel radius and distance between the rear wheels constant with known signs. The parameters are supposed uncertain with bounded uncertainties:

$$m=\hat{m}+\tilde{m},\ I=\hat{I}+\tilde{I},\ b=\hat{b}+\tilde{b},\ r=\hat{r}+\tilde{r}$$

where $\hat{m},\hat{I},\hat{b},\hat{r}$ are nominal values of mass, Inertia of moment, distance between center position and symmetry axis of mobile platform and wheel radius, and $|\tilde{m}|\le m_{\max}$, $|\tilde{I}|\le I_{\max}$, $|\tilde{b}|\le b_{\max}$, $|\tilde{r}|\le r_{\max}$ with $m_{\max},I_{\max},b_{\max},r_{\max}$ being known constants.

Also, the input disturbances are assumed to be bounded as $|\tau_{d1}|\le\tau_{d1\max}$ and $|\tau_{d2}|\le\tau_{d2\max}$ and $\tau_{d1\max},\tau_{d2\max}$ are known constants.

Let it be defined as $\beta_1=\frac{1}{b_1}$ and $\beta_2=\frac{1}{b_2}$.

The above assumptions on the uncertainty bounds for $m,b,r,I,\tau_{d1},\tau_{d2}$ produce corresponding bounded variations on the following β_1 and β_2:

$$\beta_1=\hat{\beta}_1+\tilde{\beta}_1,\quad \beta_2=\hat{\beta}_2+\tilde{\beta}_2,\quad |\tilde{\beta}_1|\le\beta_{1\max}\ \text{and}\ |\tilde{\beta}_2|\le\beta_{2\max}$$

Our objective is finding a robust feedback controller for the dynamic model Eq. (1), guaranteeing the tracking of a reference trajectory $z_r=[x_r\,y_r\,\phi_r]^T$ with asymptotically vanishing tracking errors even in the presence of plant uncertainties and input disturbances.

Let's define the tracking error as follows:

$$e=\begin{bmatrix}e_1\\ e_2\\ e_3\end{bmatrix}=\begin{bmatrix}x-x_r\\ y-y_r\\ \phi-\phi_r\end{bmatrix} \tag{2}$$

The second derivative of error is as follows:

$$\ddot{e}=\begin{bmatrix}-\frac{\lambda}{m}sin(e_3+\phi_r)+b_1(u_1+d_1)\cos(e_3+\phi_r)-\ddot{x}_r\\ \frac{\lambda}{m}cos(e_3+\phi_r)+b_1(u_1+d_1)\sin(e_3+\phi_r)-\ddot{y}_r\\ b_2(u+d_2)-\ddot{\phi}_r\end{bmatrix} \tag{3}$$

3.2 Robust siding mode controller design

Let's define two auxiliary variables as follows:

$$p_1 = \dot{x}_r - \gamma_1 e_1, p_2 = \dot{y}_r - \gamma_2 e_2 \text{ with } \gamma_1, \gamma_2 > 0 \quad (4)$$

The sliding surface s_1 is defined as

$$s_1 = \sqrt{\dot{x}^2 + \dot{y}^2} - \sqrt{p_1^2 + p_2^2} = 0 \quad (5)$$

The achievement of a sliding motion on s_1 implies the following:

$$\sqrt{\dot{x}^2 + \dot{y}^2} = \sqrt{p_1^2 + p_2^2} \quad (6)$$

The time derivative of s_1 is as follows

$$s_1 = b_1 u_1 - f(z,t) \quad (7)$$

where $f(z,t) = \dfrac{p_1 \dot{p}_1 + p_2 \dot{p}_2}{\sqrt{p_1^2 + p_2^2}}$

Similarly, $\dot{\omega} = \ddot{\phi} = b_2 u_2 = \dfrac{u_2}{\beta_2}$

In the nominal case, the control law is $u_1^{eq} = \dfrac{f(z,t)}{\hat{b}_1} = \hat{\beta}_1 f(z,t)$. During sliding motion, the reachability condition gives the control input $u_1 = u_1^{eq} + u_1^n + d_1$ that is $s_1 \dot{s}_1 \leq 0$.

$$\begin{aligned} s_1 \dot{s}_1 &= s_1 (b_1 u_1 - f(z,t)) \quad (8) \\ &= \frac{s_1}{\beta_1}[u_1^n + d_1) - \widetilde{\beta}_1 f(z,t)] \leq 0 \end{aligned}$$

To satisfy the inequality condition Eq. (8), the control input can be chosen as

$$\begin{aligned} & u_1^n = -\eta_1 (d_{1\max} + \beta_{1\max}) | f(z,t) |) sign(s_1) \\ & with \quad \eta_1 > 0 \quad (9) \end{aligned}$$

Eq. (9) ensures Eq. (8).

The following variables are defined as follows:

$$\begin{cases} \Psi = \arctan\left(\dfrac{p_2}{p_1}\right) \\ M = \Psi - \phi \end{cases} \quad (10)$$

and the sliding surface s_2 is defined as

$$s_2 = \dot{M} + \rho M = 0, \quad \rho > 0 \quad (11)$$

When a sliding motion occurs on s_2 the ϕ tends to Ψ with time constant $\dfrac{1}{\rho}$. The asymptotic tracking of the desired orientation angle is ensured by Eq. (5) and Eq. (11).

Similarly, the mobile robot is described by the dynamic model Eq. (1) with the control law

$$u_2 = u_2^{eq} + u_2^n \quad (12)$$

Coupling with the condition of sliding motion on s_1 guarantees the asymptotic vanishing of the tracking errors with dynamics assigned by γ_1, γ_2.

In the nominal case, the equivalent control input can be derived by $\dot{s}_2 = 0$, that is,

$$\begin{aligned} \dot{s}_2 = \ddot{M} + \rho \dot{M} = (\ddot{\Psi} - \ddot{\psi}) + \rho(\dot{\Psi} - \dot{\phi}) = 0 \quad (13) \\ \Rightarrow \ddot{\phi} = \ddot{\Psi} + \rho(\dot{\Psi} - \dot{\phi}) \end{aligned}$$

$$u_2^{eq} = \widehat{\beta}_2 [\ddot{\Psi} + \rho(\dot{\Psi} - \dot{\phi})] \quad (14)$$

In the perturbed situation, the sliding mode existence condition is as follows:

$$s_2 \dot{s}_2 < 0 \quad (15)$$

$$\begin{aligned} s_2 \dot{s}_2 &= s_2 [(\ddot{\Psi} - \ddot{\psi}) + \rho(\dot{\Psi} - \dot{\phi}] \quad (16) \\ &= \frac{s_2}{\beta_2} \left\{ -(u_2^{eq} + u_2^n + d_2) + \beta_2 [\ddot{\Psi} + \rho(\dot{\Psi} - \dot{\phi})] \right\} \end{aligned}$$

$$\begin{aligned} \dot{s}_2 = \ddot{M} + \rho \dot{M} = (\ddot{\Psi} - \ddot{\psi}) + \rho(\dot{\Psi} - \dot{\phi}) = 0 \\ \Rightarrow \ddot{\phi} = \ddot{\Psi} + \rho(\dot{\Psi} - \dot{\phi}) \end{aligned}$$

with $u_2^{eq} = \hat{\beta}_2 [\ddot{\Psi} + \rho(\dot{\Psi} - \dot{\phi})]$, Combining Eqs. (15) and(16) can be rewritten as follows:

$$\frac{s_2}{\beta_2} \left[-u_2^n - d_2 + \widetilde{\beta}_2 (\ddot{\Psi} + \rho(\dot{\Psi} - \dot{\phi}) \right] < 0 \quad (17)$$

To satisfy the condition Eq. (15), the control input can be chosen as

$$u_2^n = -\eta_2 [d_{2\max} + \beta_{2\max} | \ddot{\Psi} + \rho(\dot{\Psi} - \dot{\phi}) |] sign(s_2) \quad (18)$$

with $\eta_2 > 1$

When a sliding motion occurs on Eq. (13), the vanishing of M implies that ϕ tends to Ψ, that is,

$$\tan(\phi)=\frac{p_2}{p_1} \tag{19}$$

On the other hand, the nonholonomic constraints gives

$$\begin{cases}\sin(\phi)=\dfrac{\dot{y}}{\sqrt{\dot{x}^2+\dot{y}^2}}=\dfrac{p_2}{\sqrt{p_1^2+p_2^2}}\\ \cos(\phi)=\dfrac{\dot{x}}{\sqrt{\dot{x}^2+\dot{y}^2}}=\dfrac{p_1}{\sqrt{p_1^2+p_2^2}}\end{cases} \tag{20}$$

Additionally, the simultaneous achievement of sliding motion on s_1 gives

$$\sqrt{\dot{x}^2+\dot{y}^2}=\sqrt{p_1^2+p_2^2} \tag{21}$$

From Eq. (20) and Eq. (21) it follows

$$\begin{aligned}\dot{x}&=p_1=\dot{x}_r-\gamma_1 e_1\\ \dot{x}&=p_2=\dot{y}_r-\gamma_2 e_2\end{aligned} \tag{22}$$

That is,

$$\begin{aligned}\dot{x}-\dot{x}_r&=\dot{e}_1=-\gamma_1 e_1\\ \dot{y}-\dot{y}_r&=\dot{e}_2=-\gamma_2 e_2\end{aligned} \tag{23}$$

Eq. (23) ensures that e_1 and e_2 asymptotically vanish with assigned dynamic. As a result, from Eq. (10), Ψ converges to $\arctan\left(\frac{\dot{x}_r}{\dot{y}_r}\right)$. Moreover, due to Eq. (11), ϕ converges to the desired orientation. The block diagram of the controller is given in Fig. 4.

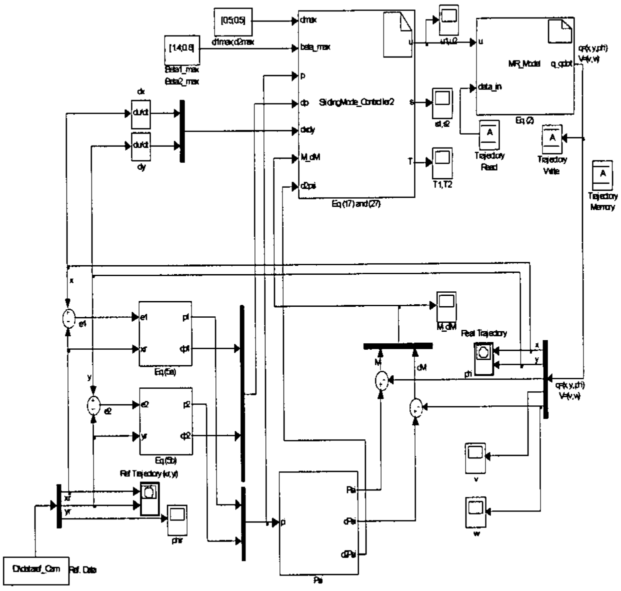


Fig. 4 Block diagram of the controller

4. Mobile Robot Localization

In this paper, the position of the mobile robot can be detected using camera. The camera calibration processing uses one of the well-established Tsai's methods which are popular among computer vision researchers: mono-view coplanar points with a known target object. With this method, the invariable intrinsic parameters of the camera can be derived, and they are used to compute the camera's localization with respect to the object based on a number of feature points on it. The camera localization is, in turn, useful information for path planning scheme and for mobile robot localization.

4.1 Camera model

The basic geometry of a pin-hole camera model is illustrated in Fig. 5. In the figure, $(O_w - X_w Y_w Z_w)$ is the 3D coordinate system of world space, and $(O - XYZ)$ is the 3D coordinate system of the camera frame with the camera origin O at the focus point of the camera and the Z-axis coinciding with the optical axis and pointing to the back of the camera. Let $(O - XY)$ be the 2D coordinate system of CCD/CMOS sensor image frame, and $(O - X_F Y_F)$ is the 2D computer image coordinate system. The distance between the sensor image frame and the camera origin is called camera focal length f.

Let $P_w = (x_w, y_w, z_w)$be a point in the 3D world space, called the feature point, and its camera coordinate representation be $P = (x, y, z)$. The coordinates (X_d, Y_d) is the sensor image coordinate of P_w. The coordinate (X_d, Y_d) is the actual image coordinate which differs from (X_u, Y_u) due to lens distortion. The coordinate (X_f, Y_f) represents the coordinate in the computer memory which can be obtained through image processing. The coordinate (C_x, C_y) represents the image center position in computer memory, and it is assumed that image center position is located at center of the computer memory.

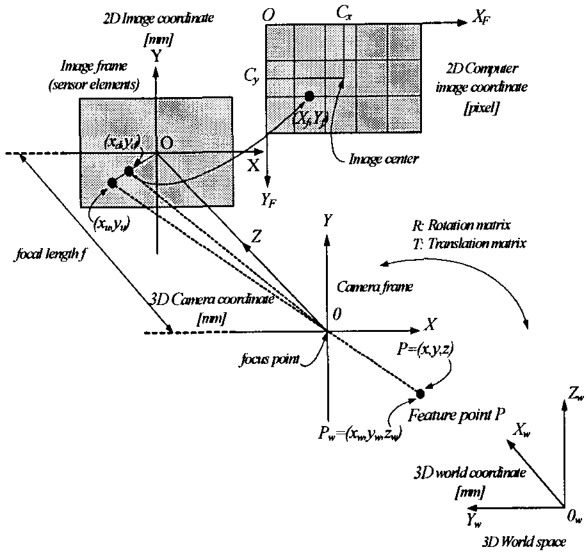


Fig. 5 Pin-hold camera model

4.2 Camera calibration using mono-view coplanar points

STAGE 1: Compute the extrinsic parameters T_x, T_y, R_x, T_y, R_2

Step 1:Compute the distorted image coordinate (X_d, Y_d)

$$\begin{cases} X_{di} = s_x^{-1} d_x' (X_{fi} - C_x) \\ Y_{di} = d_y (Y_{fi} - C_y) \end{cases} \tag{24}$$

where $i = 1 ... n$, N is the number of feature points and must be larger than five.

Step 2: Compute the five unknowns $T_y^{-1} r_1$, $T_y^{-1} r_2$, $T_y^{-1} T_x$, $T_y^{-1} r_4$, $T_y^{-1} r_5$

For each point i with (x_{wi}, y_{wi}, z_{wi}) and (X_{di}, Y_{di}) in the above, the following linear equation must be setup with $T_y^{-1} r_1, T_y^{-1} r_2$, $T_y^{-1} T_x$, $T_y^{-1} r_4$, and $T_y^{-1} r_5$ as unknowns:

$$[\,Y_{di} x_{wi} \quad Y_{di} y_{wi} \quad Y_{di} \quad -X_{di} x_{wi} \quad -X_{di} y_{wi}\,] \begin{bmatrix} T_y^{-1} r_1 \\ T_y^{-1} r_2 \\ T_y^{-1} T_x \\ T_y^{-1} r_4 \\ T_y^{-1} r_5 \end{bmatrix} = X_{di}$$

(25)

An over-determined system of linear equations

Eq. (25) can be solved for the five unknowns $T_y^{-1}r_1, T_y^{-1}r_2$, $T_y^{-1}T_x$, $T_y^{-1}r_4$, and $T_y^{-1}r_5$.

Step 3: Compute $(r_1,...r_9, T_x, T_y)$ from $(T_y^{-1}r_1, T_y^{-1}r_2, T_y^{-1}T_x, T_y^{-1}r_4, T_y^{-1}r_5)$

STAGE 2: Compute effective focal length f, distortion coefficients k_1, and T_Z

Step 4: Compute an approximation of f and T_Z by ignoring lens distortion:

For each feature point i, the following linear equation is established with f and T_Z as unknowns:

$$[y_i \ -d_y Y_i]\begin{bmatrix} f \\ T_Z \end{bmatrix} = w_i d_y Y_i \qquad (26)$$

where

$$y_i = r_4 x_{wi} + r_5 y_{wi} + r_6 \cdot 0 + T_y$$

$$w_i = r_7 x_{wi} + r_8 y_{wi} + r_9 \cdot 0$$

Since rotation matrix R, translations T_x and T_y are known, this yields an over-determined system of linear equations that can be solved for the unknowns f and T_Z.

Step 5: Compute the exact solution for f, T_Z, k_1

Eq. (26) is solved with f, T_Z, k_1 as unknowns using standard optimization scheme such as steepest descent. The approximation for f and T_Z is used which are computed in step 4 as initial guess, and zero as the initial guess for k_1. The camera calibration test bed and computer user interface are shown in Fig. 6 and 7. The results of the camera parameters are given in Table 1.

Table 1 Camera parameters and calibration results

Calibration Results	
Focal Length f	7.96 [mm]
Lens Distortion Coefficient k_1	0.003946 [1/mm^2]
X Translation T_x	1.49 [mm]
Y Translation T_y	2.84 [mm]
Z Translation T_z	429.38 [mm]
X Rotation R_x	5.98 [deg]
Y Rotation R_y	8.73 [deg]
Z Rotation R_z	-0.02 [deg]

Rotation Matrix
$R = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix}$
	0.988412 0.016252 0.150921
	-0.000427 0.994545 -0.104303
	-0.151792 0.103030 0.983028

Distorted image plane error	
Error Mean	2.1565 [pixel]
Standard Deviation	0.5872 [pixel]
Error Max	2.9780 [pixel]
Undistorted image plane error	
Error Mean	2.1432 [pixel]
Standard Deviation	0.5804 [pixel]
Error Max	2.9322 [pixel]

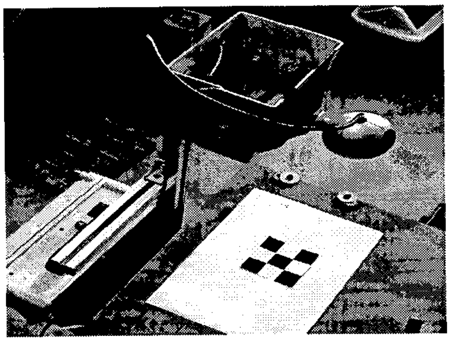


Fig. 6 Camera calibration test bed

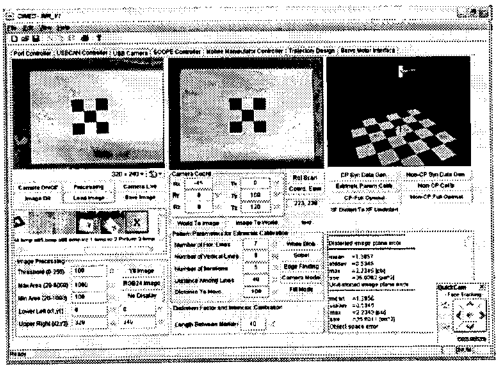


Fig. 7 Camera user interface

5. Development of Control System

The control system is based on the integration of computer and PIC-based microprocessor. The computer functions as high as high level control for image processing and control algorithm, and the microprocessor, as low level controller for device control. The configuration diagram of the overall control system is shown in Fig. 8 and the control system is given in Fig. 9, respectively.

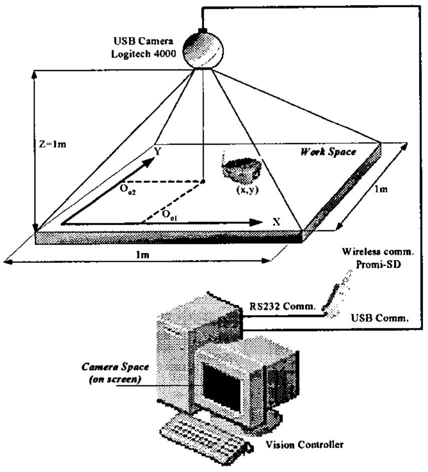


Fig. 8 The work space of the total system

For the operation, USB camera Logitech 4000 is used to capture the image stream into memory with size of 32.0×240 at 30fps using QuickCam SDK. The image is processed using image processing library Open CV to extract the features from the image for the object's position detection. The torque command is sent to the low level to control the mobile robot to track a certain trajectory.

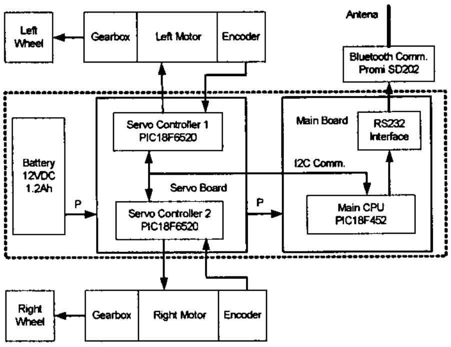


Fig. 9 Diagram of the control system

6. Simulation Results

To verify the effectiveness of the controller, the simulations have been done with controller Eq. (12) and Eq. (21) using Simulink. The reference trajectory for the mobile platform is shown in Fig. 10. The mobile robot is described by the dynamic model Eq. (4) with the following nominal values for the physical parameters:

$m = 20.6Kg$, $I = 2Kgm^2$, $r = 110mm$, $R = 200mm$. The initial values are $x(0) = 0.28m$, $y(0) = 0.395m$, and $\phi(0) = 5^\circ$. The controller are set as follows:$r_1 = 5$, $r_2 = 5$, $\rho = 7.5$, $\eta_1 = \eta_2 = 7.5$. The constant input disturbances $d_1 = 0.3Nm$ and $d_2 = 0.3Nm$ has been considered with the maximum amplitude, $d_{1\max} = d_{2\max} = 0.5Nm$ but they are not considered in this simulation.

Simulation results are given through Figs. 11-19. The mobile platform's tracking errors are given in Fig. 11 for full time (20s) and Fig. 12 for the initial time (4s), respectively. It can be seen that the platform errors go to zero after about 3 seconds. The sliding surface s_1 and s_2 are shown in Figs.13 and 14, respectively. The sliding surfaces are chattering about the values of zero as desired and continue sliding on that surfaces. The linear and angular velocities at mobile platform center are shown in Figs. 15 and 16 for full time (20s). The velocities of the left and right wheel are shown in Figs. 18. It can be seen that the tracking velocity is in the vicinity of 100mm/s as desired. Fig. 19 shows the velocities of the wheels for part time(6s).

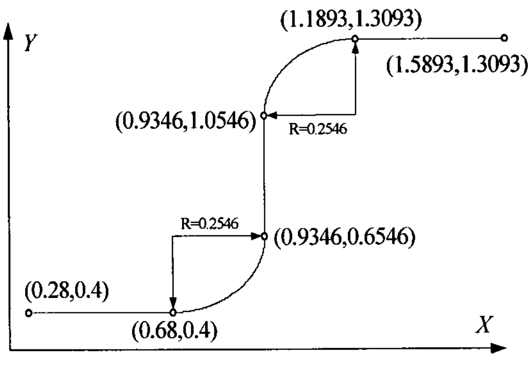


Fig. 10 Reference trajectory

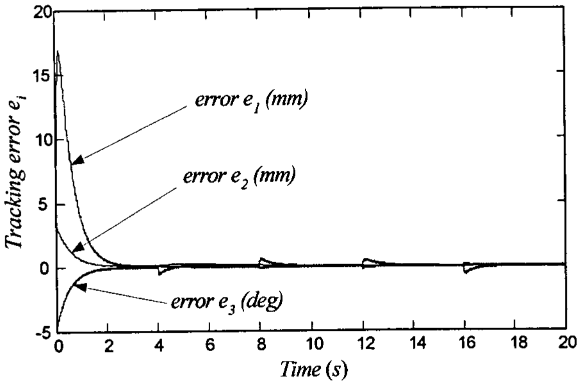


Fig. 11 Platform tracking errors for full time

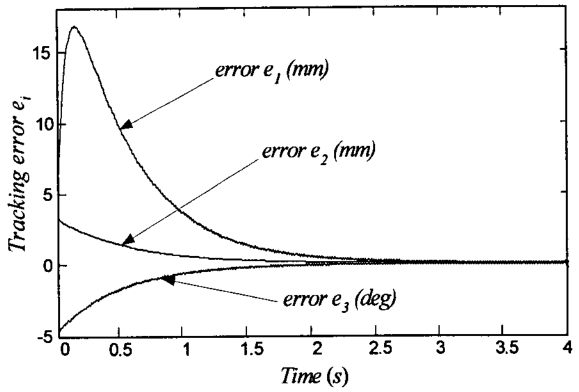


Fig. 12 Platform tracking errors for the initial time

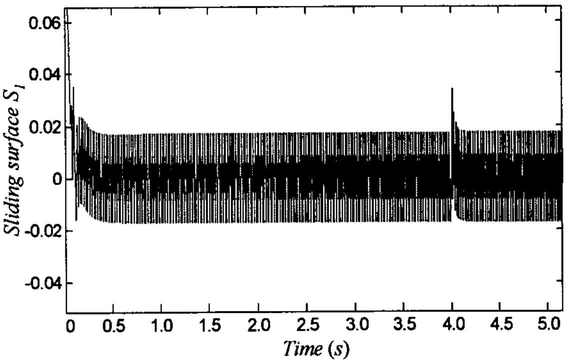


Fig. 13 Sliding surface s_1

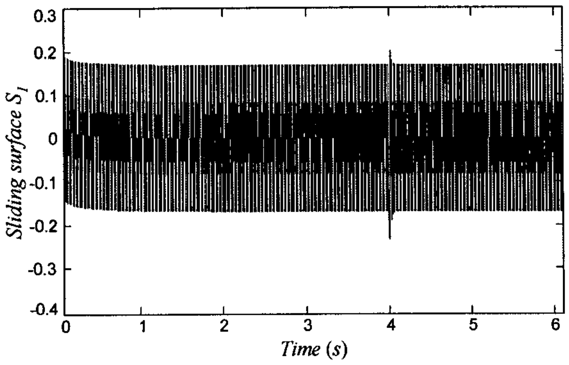


Fig. 14 Sliding surface s_2

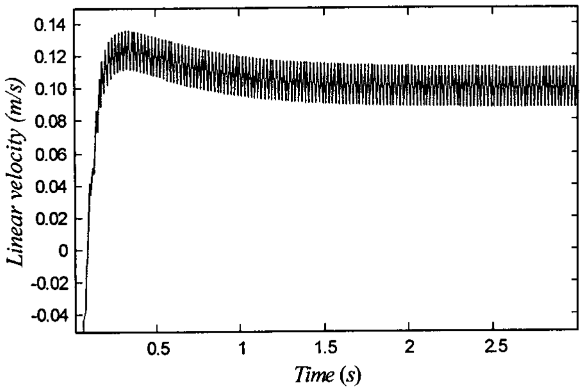


Fig. 15 Linear velocity at MR's center

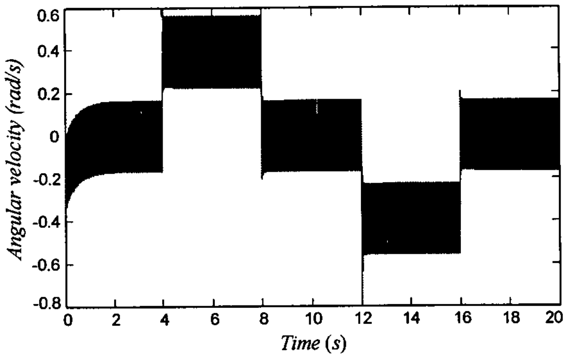


Fig. 16 Angular velocity

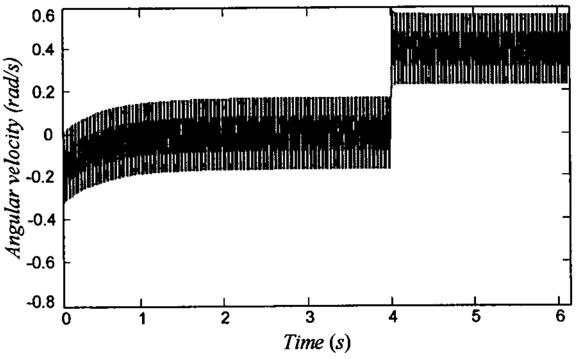


Fig. 17 Angular velocity for part time (6s)

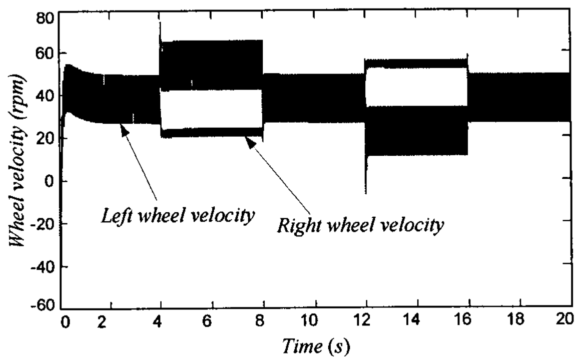


Fig. 18 Velocities of the left and right wheel

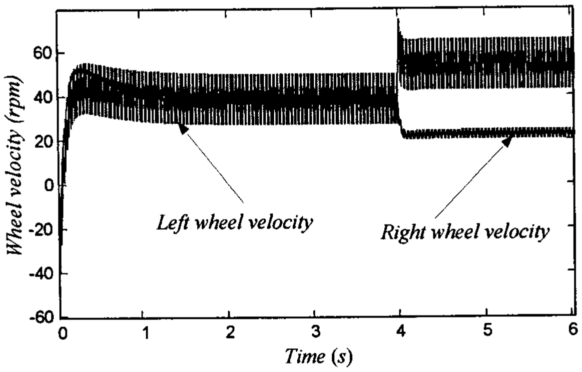


Fig. 19 Velocities of the wheels for part time (6s)

7. Conclusions

In this paper, the robust tracking problem for the dynamical model of a WMR is solved using sliding mode control. The presence of bounded parameter uncertainties and input disturbances are considered. The controller is designed based on two nonlinear sliding surfaces ensuring the tracking of the three output variables exploiting the nonholonomic constraint. The robust asymptotic vanishing of the tracking errors has been theoretically proved using sliding mode control. Also, the shortest path finding based on Dijkstra's algorithm and the camera calibration results have been verified.

References

1. D. K. Chwa, J. H. Seo, P. Kim and J. Y. Choi, 2002, “Sliding Mode Tracking Control of Nonholonomic Wheeled Mobile Robots”, Proceedings of the American Control Conference Anchorage, pp. 3991-3996.
2. T. Fukao, H. Nakagawa and N. Adachi, 2000, “Adaptive Tracking Control of a Nonholonomic Mobile Robot”, IEEE Transaction on Robotics and Automation, Vol. 16, No. 5, pp. 609-615.
3. A. Hoover and B. D. Olsen, 1999, “Path Planning for Mobile Robots Using a Video Camera Network”, IEEE Int. Conf. Adv. Intell. Mech., Atlanta, USA.
4. R. Fierro and F.L. Lewis, 1995, “Control of a Non-holonomic Mobile Robot: Backstepping Kinematics into Dynamics”, Proceedings of IEEE Conference on Decision & Control, pp. 3805-3810.
5. S. Sundar and Z. Shiller, 1997, “Optimal Obstacle Avoidance Based on the Hamilton-Jacobi-Bellman Equation”, IEEE Trans. Robot. Automat, Vol. 13, pp. 305-310.