Optical flow based odometry for mobile robots supported by multiple sensors and sensor fusion

This paper introduces an optical flow based odometry solution for indoor mobile robots. The indoor localization of mobile robots is an important issue according to the increasing mobile robot market and the needs of the industrial, service and consumer electronics sectors. The robot odometry calculated from the robot kinematics accumulates the position error caused by the wheel slip but an optical flow based measurement is independent from wheel slipping so both methods have different credibility which was considered during the sensor fusion and the development. The focus of the research was to design an embedded system with high accuracy on the possibly lowest price to serve the needs of the consumer electronics sector without the need of expensive camera and real-time embedded computer based high level robot localization solutions. The paper proposes the theoretical background, the implementation and the experimental results as well. The universal optical flow module can be implemented in any kind of indoor mobile robot to measure the position and the orientation of the robot during the motion, even in the case of a 3 DoF holonomic drive like kiwi drive. The application of omnidirectional wheels in mobile robotics requires high accurate position and orientation feedback methods contrary to differential drives.

Key words: Mobile robot, Omni drive, Odometry, Optical flow, Sensor correlation

1 INTRODUCTION

The need for automation and robotics solutions is an everyday topic in the industrial sector. There is also a growing need in daily life situations, such as floor cleaning, lawn moving, window cleaning and other things solvable by mobile robotics. The motion in an indoor or an outdoor environment is the most important task of a mobile robot, which is impossible without appropriate localization. The required accuracy of the robot position and orientation is usually defined by the application. The accuracy requirements are different in case of outdoor and indoor mobile robots, where the travel distance is also different. As an example, 2 meters inaccuracy is acceptable during a 250 meters quad copter flight, but in case of an indoor robot the width of a corridor can be only 2 meters. For indoor localization, several vision and kinematics based methods and applications exist like [1, 2]. The kinematics based solution calculates the robot position and orientation from the angular position of the wheels, which accumulates the error according to the slippery floor [3-6]. The vision based methods requires embedded computers or normal external PC-s to provide enough calculation throughput for the al-
Optical flow based odometry for mobile robots supported by multiple sensors and sensor fusion

F. Tajti, G. Szayer, B. Kovács, P. Barna, P. Korondi

Algorithms [7]. Our solution for the improvement of wheel based odometry is an optical based method with minimal calculation throughput on the lowest price for the mobile robots of the service sector. Service robot usually does not have any kind of embedded computers, but has simple RISK microcontroller based embedded systems. It is also important to mention; that the proposed method does not need the use of any external localization elements, like infra light beacons, ultrasonic sensors, visual markers, etc. In the consumer electronics sector the production of mobile robots should be solved on the possibly lowest variable costs. Difficult software based solutions with cheap hardware elements are beneficial for products manufactured in quantity.

At first this paper describes the control problems of omnidirectional mobile robots. In the next section three methods are proposed for measuring position and orientation. Section 3 describes the odometry based methods. Section 4 describes the fusion of the results given from the different methods, then section 5 describes the open loop test results. Finally, section 6 concludes the paper.

2 ODOMETRY INACCURACY

In the view of the motion on the ground plane a mobile robot can have 2 or 3 degrees of freedom (DoF) depending on the structure of the drive. In case of 3 DoF (e.g. holonomic drives), the robot can change the position and the orientation at the same time, but a 2 DoF robot (e.g. differential drives) cannot change position and orientation independently. A mobile robot with holonomic drive can also move in any ways on the ground plane without the change of the robot orientation. Holonomic robots have mechanisms to control all 3 physical DoF on the ground plane. Holonomic drives with three omniwheels are also known as kiwi drives. The research was inspired by Ethon (see Fig. 1) [8], which is our self-designed kiwi drive based holonomic mobile robot, made for ethological research. On a slippery floor after 15-30 meters of motion in different directions the odometry of Ethon accumulated 2-5 meter errors in any directions and 10-30° in orientation, which is impermissible. [9] The errors of the position and the orientation come from the following problems:

- the wheels are slipping [10,11]
- the contact points between the wheels and the floor alternate (see Fig. 2)
- the results of the sine and cosine are calculated from a 1° resolution look-up table
- the sine and the cosine functions in the implementation of the inverse kinematics are formulated to tangent functions and estimated with the division of integer numbers

The error related to the contact points (δ₁) is 5.4%, which can be calculated as (1), where Δ(W₁,W₂) is the distance between the contact points of wheel 1 and r (the radius of the robot geometry) is the distance between the center of the robot structure and the wheels.

\[ \delta_1 = \frac{\Delta (W_1,W_2)}{r} = \frac{16.2 \text{ mm}}{300 \text{ mm}} = 0.054 \]  

(1)

The 1° resolution look-up table causes 0.27% error (δ₂). The estimation of the tangent function causes 2.15% error (δ₃). According to the errors (δ₁, δ₂, δ₃) and the undefined wheel slipping error (δ₄) the robot accumulates in the worst case about ±5-8 % error. During the measurements at 15 meters of continuous linear driving in the robot position it caused in worst case 0.5-1 meter error. This can be accepted and compensated with the high level mapping localization of the robot. During the measurements of the angular position the ±5-8% position error caused about 10-30° angular error between the robot coordinate system and the world coordinate system. If the robot moves sideways with 10-30° angular error it causes 11-33% position error, which cannot be accepted, because it means 5 meters in a
15 meter drive. (In case of 90° error the robot changed the x-y directions, which is 100% position error, because sin 90° = 1.)

3 ODOMETRY METHODS

Mobile robots can be controlled without position and orientation feedback. In this case only the servo amplifiers have their own angular position feedback during the motion and the references of the wheels can be calculated by the equations of the inverse kinematics. For a mobile robot with differential drive, this method provides an appropriate indoor localization. In case of mobile robot drives, where the wheels or tracks slip on the ground more sophisticated control methods are required. To solve the problems of the robot localization three types of measurement methods were implemented and investigated on the robot: 3/3/3 sensor, ADNS9500 optical flow sensors, and inverse ge-}

3.1 Odometry based on wheel rotation

For the motion control of the robot, both the inverse kinematics and the direct geometry were implemented in the central controller unit of the robot, which provides the wheel references for the servo amplifiers of the robot real-time in 1 kHz via CAN-bus interface. The transformation between the world coordinate system and the robot coordinate system can be described as (2) with one rotational (around axis z) and two linear transformations (along x and y). (See Fig. 4.)

where

(2)

The transformation of a point from world coordinate system to robot coordinate system can be expressed straightforward as (3).

(3)

Where the offset can be expressed as a simple vector (4).

(4)

During the implementation of the motion control methods dimensionless integer numbers were used by calculation throughput optimising considerations. The references of the wheels can be expressed from the robot references as (3.5-3.7), where K and C are constants to get real units.

(6)

(7)

From the equations of the inverse kinematics (3.5-3.7), the equations of the direct geometry can be expressed as (3.8-3.10), where \( \varphi_0, x_0, y_0 \) are the initial position and orientation of the robot and \( \varphi, x, y \) are the actual position and orientation of the robot.

(9)

(10)

(11)

During the robot motion, both the position and the orientation values can be updated externally (for example from
sensor fusion) in every 50 ms (20 Hz), where at least the orientation should be updated. The motion control of the robot is calculated at 1 kHz, which is 1 ms. According to the maximal velocity (1.5 m/s) and the maximal angular velocity (1.47 rad/s) of the robot the system will not accumulate too much error between two position updates (50 ms) so the motion control can calculate 50 cycles with the values of the inverse geometry functions.

3.2 Gyroscope, accelerometer and magneto sensor

As it is mentioned above, the odometry calculated from equations of the direct geometry accumulated 33% orientation error and only 8% position error. PhidgetSpatial Precision 3/3/3 sensor were used to compensate the orientation problems, which is a measurement card with a 3 axis accelerometer, a 3 axis gyroscope and a 3 axis magnetometer [12]. For PhidgetSpatial Precision 3/3/3 several open source implementations exist. On Ethon, we used sensor fusion with Kalman filter for the gyroscope, pitch and roll calculation and finally tilt compensation to get the angular position value (\(\dot{\varphi}\)), which is the tilt angle (\(\theta\)) (see Fig. 5). The measurement card and the implemented method could provide 0.01 radian resolution, which means only 0.16% angular error.

3.3 Optical flow

For the optical flow method we used ADNS9500 gaming mouse sensors, which can measure motion in two directions from 5-10 mm floor height. It is an important fact that different floor surface characteristics influence the measurement. The ADNS9500 sensor measures the surface quality during motion, which can be read from SQUAL sensor register. The maximal value of the register is 169. A white paper can reach around 40 and the register value is nearly equal to zero when there is no surface below the sensor. The value of the SQUAL register was used during the calculation of the optical flow sensor credibility.

The robot has 3 DoF so the system requires two sensors to be able to measure both the robot position and orientation [13-15]. Between the sensors and the embedded system we implemented SPI (serial peripheral interface) communications. The position and the orientation of the sensors in robot coordinate system can be described with the following parameters: \(x_A, y_A, z_A, r_A0, \varphi_A0\) and \(x_B, y_B, z_B, r_B0, \varphi_B0\) - as figure 5, where \(r_A0, \varphi_A0\) and \(r_B0, \varphi_B0\) are constants. From the optical flow sensors we used only raw data without any ergonomics functions, like mouse acceleration which is usually implemented in computer peripherals. The start of the measurement is synchronized between the sensors in the embedded system and the software updates the absolute \(x_A, y_A, x_B, y_B\) and the \(\Delta x_A, \Delta y_A, \Delta x_B, \Delta y_B\) values in discrete time so the equations and the implementation of the inverse kinematics [16, 17] are based on these values. (See Fig. 6.)

The \(\Delta x_A, \Delta y_A, \Delta x_B, \Delta y_B\) Parameters of the measurement can be described as (12)-(15).

\[
\begin{align*}
\Delta x_A &= dx_w c(\varphi_A0 + \varphi) + dy_w s(\varphi_A0 + \varphi) + d\varphi(x_{A0}s\varphi_A0 - y_{A0}c\varphi_A0) \\
\Delta y_A &= -dx_w s(\varphi_A0 + \varphi) + dy_w c(\varphi_A0 + \varphi) + d\varphi(x_{A0}c\varphi_A0 + y_{A0}s\varphi_A0) \\
\Delta x_B &= dx_w c(\varphi_B0 + \varphi) + dy_w s(\varphi_B0 + \varphi) + d\varphi(x_{B0}s\varphi_B0 - y_{B0}c\varphi_B0) \\
\Delta y_B &= -dx_w s(\varphi_B0 + \varphi) + dy_w c(\varphi_B0 + \varphi) + d\varphi(x_{B0}c\varphi_B0 + y_{B0}s\varphi_B0)
\end{align*}
\]
Equations (12)-(15) can be described as (16).

\[
\begin{bmatrix}
  c(\varphi_A + \varphi) & s(\varphi_A + \varphi) & x_{A}A s\varphi_A - y_{A}A c\varphi_A \\
  -s(\varphi_A + \varphi) & c(\varphi_A + \varphi) & x_{A}A c\varphi_A + y_{A}A s\varphi_A \\
  c(\varphi_B + \varphi) & s(\varphi_B + \varphi) & x_{B}B s\varphi_B - y_{B}B c\varphi_B \\
  -s(\varphi_B + \varphi) & c(\varphi_B + \varphi) & x_{B}B c\varphi_B + y_{A}B s\varphi_B
\end{bmatrix}
\begin{bmatrix}
  dx_w \\
  dy_w \\
  d\varphi
\end{bmatrix}
= 
\begin{bmatrix}
  dx_A \\
  dy_A \\
  dx_B \\
  dy_B
\end{bmatrix}
\tag{16}
\]

In this case we measure a 3 DoF system with four parameters, which are not independent and one parameter can be calculated from the others to implement data correlation. The fourth parameter can be expressed from the others generally as (17) and (18).

\[
A = \begin{bmatrix} x_{A}A - x_{B}B \\ y_{A}A - y_{B}B \end{bmatrix}
\tag{17}
\]

\[
y_B = \frac{x_A + \sqrt{3} y_A + 2 x_B}{\sqrt{3}}
\tag{18}
\]

In our special case points A and B lie in \(r_A\) radius with 120° orientation, where \(x_A, x_B\) in the line of the radius and \(y_A, y_B\) are perpendicular to the radius. With these conditions \(y_B\) can be expressed as (19) and (20).

\[
y_B = -1/\sqrt{3} (x_A - x_B) - y_A
\tag{19}
\]

\[
y_B = 1/\sqrt{3} (x_A - x_B) - y_A
\tag{20}
\]

According to our special case and \(r_A, \varphi_A, r_B, \varphi_B\) constant parameters (16) can be expressed as (22). During the implementation of the system on different robots other simplifications can be made in (16) according to the actual position and orientation of the optical flow sensors in the robot coordinate system.

\[
\begin{bmatrix}
  dx_A \\
  dy_A \\
  d\varphi
\end{bmatrix}
= 
\begin{bmatrix}
  c(\varphi_A + \varphi) & s(\varphi_A + \varphi) & -r_0 \\
  -s(\varphi_A + \varphi) & c(\varphi_A + \varphi) & 0 \\
  -s(\varphi_B + \varphi) & c(\varphi_B + \varphi) & 0
\end{bmatrix}
\begin{bmatrix}
  dx_w \\
  dy_w \\
  d\varphi
\end{bmatrix}
\tag{21}
\]

The pseudo inverse of (21) gives the solution for the Jacobian matrix of the system as (22).

\[
\begin{bmatrix}
  dx_w \\
  dy_w \\
  d\varphi
\end{bmatrix}
= 
\begin{bmatrix}
  -\sqrt{3} \varphi_A + \frac{\sqrt{3}}{\sqrt{3}} & -\sqrt{3} \varphi_B + \frac{\sqrt{3}}{\sqrt{3}} & \frac{\sqrt{3}}{3} \varphi_A - \frac{z_0}{3r_0} \\
  -\sqrt{3} \varphi_B + \frac{\sqrt{3}}{\sqrt{3}} & -\sqrt{3} \varphi_A + \frac{\sqrt{3}}{\sqrt{3}} & \frac{\sqrt{3}}{3} \varphi_B - \frac{z_0}{3r_0}
\end{bmatrix}
\begin{bmatrix}
  dx_A \\
  dy_A \\
  dx_B \\
  dy_B
\end{bmatrix}
\tag{22}
\]

In this case the Jacobian matrix can be separated for a rotational matrix around axis \(z\) and a transformational matrix on the \(x-y\) plane as (23).

\[
J = \begin{bmatrix}
  c\varphi & -s\varphi & 0 \\
  s\varphi & c\varphi & 0 \\
  0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
  0 & -\frac{\sqrt{3}}{3} & \frac{\sqrt{3}}{3} \\
  \frac{\sqrt{3}}{3} & 0 & \frac{\sqrt{3}}{3} \\
  -\frac{\sqrt{3}}{3} & -\frac{\sqrt{3}}{3} & \frac{\sqrt{3}}{3}
\end{bmatrix}
\tag{23}
\]

Equation (23) can be expressed in complex form as (24). (24)

\[
\begin{bmatrix}
  dx_A \\
  dy_A \\
  dx_B \\
  dy_B
\end{bmatrix}
= 
\begin{bmatrix}
  e^{-i\varphi} & 0 & 0 \\
  0 & e^0 & 1 \\
  -\frac{1+i\sqrt{3}}{3} & -\frac{2}{3} & 0 \\
  -\frac{1+i\sqrt{3}}{3} & -\frac{2}{3} & 0
\end{bmatrix}
\begin{bmatrix}
  dx_w + idy_w \\
  -\sqrt{3} \varphi_A + \frac{z_0}{3r_0} \\
  \frac{\sqrt{3}}{3} \varphi_A - \frac{z_0}{3r_0} \\
  \frac{\sqrt{3}}{3} \varphi_B - \frac{z_0}{3r_0}
\end{bmatrix}
\tag{25}
\]

From (24) the following conclusions can be made:

1. \(\varphi\) (robot orientation) is independent from the length of the path and can be expressed from the actual values of the optical flow sensors

2. to express \(x, y\) values (robot position) first the calculation of \(\varphi\) is needed

Simplifying (24) with the value of \(y_B\) we can describe the equation as (25).

\[
\begin{bmatrix}
  dx_w + idy_w \\
  -\sqrt{3} \varphi_A + \frac{z_0}{3r_0} \\
  \frac{\sqrt{3}}{3} \varphi_A - \frac{z_0}{3r_0} \\
  \frac{\sqrt{3}}{3} \varphi_B - \frac{z_0}{3r_0}
\end{bmatrix}
= 
\begin{bmatrix}
  e^{-i\varphi} & 0 & 0 \\
  0 & e^0 & 1 \\
  -\frac{1+i\sqrt{3}}{3} & -\frac{2}{3} & 0 \\
  -\frac{1+i\sqrt{3}}{3} & -\frac{2}{3} & 0
\end{bmatrix}
\begin{bmatrix}
  dx_A \\
  dy_A \\
  dx_B
\end{bmatrix}
\tag{25}
\]

From (25) \(y_w\) (the \(y\) coordinate of the robot position) can be expressed as (26), \(x_w\) (the \(x\) coordinate of the robot position) can be expressed as (27) and \(\varphi\) (the orientation of the robot) can be expressed as (28).

\[
x_w = \int_0^t \left( \frac{x_A}{3} - \frac{2 y_A}{\sqrt{3}} - \frac{x_B}{3} \right) \left( \frac{z_0 + \sqrt{3} y_A + 2 x_B}{3r_0} \right) + \frac{\sqrt{3} \varphi}{3r_0} \left( x_A - x_B \right) c \left( \frac{z_0 + \sqrt{3} y_A + 2 x_B}{3r_0} \right) \right) dt
\tag{26}
\]

\[
y_w = \int_0^t \left( \frac{\sqrt{3}}{3} \left( x_A - x_B \right) \left( \frac{z_0 + \sqrt{3} y_A + 2 x_B}{3r_0} \right) - \frac{2 y_A}{\sqrt{3}} - \frac{x_B}{3} \right) \left( \frac{z_0 + \sqrt{3} y_A + 2 x_B}{3r_0} \right) \right) dt
\tag{27}
\]

\[
\varphi = -\frac{x_A + \sqrt{3} y_A + 2 x_B}{3r_0}
\tag{28}
\]
With (26), (27) and (28) the homogenous transformational matrix of the system can be expressed as (29).

\[
H = \begin{bmatrix}
c\varphi & -s\varphi & 0 & x_w \\
s\varphi & c\varphi & 0 & y_w \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\] (29)

For the validation of the equations of the optical flow embedded system we used our open source real-time Linux (and LinuxCNC) based robot controller. During the measurements a SEIKO D-Tran 4400 SC SCARA robot moved the sensor system in 3 DoFs \((x, y, \varphi)\). The accuracy of the robot is higher with a magnitude than the accuracy of the optical flow system. In case of this robotized validation measurement, the position and the orientation can be modified accurately and the measurement can be repeated. See Fig. 7 and Fig. 8. Refocusing the sensors the method can be improved [18, 19]. The first measurements were made with constant orientation at each point in order to simulate holonomic movement. The grey line on Fig. 9. was the path of the robot, which was plotted from G-code and the dotted black line was measured with the optical flow system. The difference between the lines (the position error) is acceptable (less than 10 mm). During second measurement the orientation was always perpendicular to the direction of motion (like the motion of a mobile robot with differential drive). The grey line represents the path of the robot (plotted from the G-code) and the black dotted line represents the path, which was measured by the optical flow sensors. See Fig. 10. In this case the error is also acceptable (less than 10 mm). The optical flow system measures the motion of its own center point and it could not be fixed to the tool center point (TCP) of the SCARA robot with less than 2-3 mm eccentricity. This offset caused about 5 mm error at the change of the orientation on the edges of the rectangle, because in this case the TCP of the optical flow moved on an arc. See Fig. 11.

4 SENSOR CORRELATION AND CREDIBILITY

The motion of the robot can be measured with three methods as it is described above. The proposed methods and the sensor fusion are implemented in an embedded system to give position and orientation feedback for the robot control. In this method the credibility is the weight of the measured parameter in the equations of the sensor fusion. The block diagram of the implementation can be seen at Fig. 12, where the parameters are the following:

1. \(x_{\text{ref}}, y_{\text{ref}}, \varphi_{\text{ref}}\) are the position and angular position references in robot coordinate system
2. \(x, y, \phi\) are the position and the angular position of the robot

3. \(\hat{x}, \hat{y}, \hat{\phi}\) are the estimated position and the estimated angular position of the robot

4. \(\omega_1, \omega_2, \omega_3\) are the angular positions of the wheels

5. \(\ddot{x}, \ddot{y}, \ddot{\phi}\) are the accelerations of the robot

6. \(x, y, \phi\) are the position and the angular position of the robot

7. \(\rho, \phi, \theta\) are the angular position of the robot

8. \(\dot{\rho}, \dot{\phi}, \dot{\theta}\) are the angular velocities of the robot

9. \(\Delta x_1, \Delta y_1, \Delta x_2, \Delta y_2\) are the position changes in sensor coordinate systems

10. \(h\) is the sensor height and \(\hat{h}\) is the credibility of the odometry sensor

11. \(\hat{x}_\rho, \hat{y}_\rho, \hat{\phi}_\rho\) are the position and the angular position of the robot calculated from the inverse kinematics

12. \(\hat{x}_\sigma, \hat{y}_\sigma, \hat{\phi}_\sigma\) are the position and the angular position of the robot calculated from the optical flow sensors

13. \(\hat{\phi}_\zeta\) is the estimated angular position of the robot calculated from the 3/3/3 sensor fusion

14. \(\hat{\zeta}\) is a credibility parameter estimated by the inverse kinematics

The mobile robot can be controlled similarly to many industrial machines like CNC-s, milling machines and manipulators, where the path planning method calculates the references for the servo amplifiers and there is no feedback for the central controller. In this configuration -in case of any fault in a joint- the servo amplifier generates an error signal and the central controller disables the whole machine. The designed embedded system is synchronized with the real-time motion controller of the robot and provides position and angular position feedback for the system. The motion control and the position feedback runs at 1 kHz cycle time so the time constant of the controller is much lower than the mechanical time constant. The \(\hat{x}_\rho, \hat{y}_\rho, \hat{\phi}_\rho, \hat{\zeta}\) values of the inverse kinematics module can be calculated on this frequency, where \(\hat{\zeta}\) is an estimated credibility parameter, which comes from the unexpected events of the motion like follow error, motor current, wheel accelerations \((\ddot{\omega}_1, \ddot{\omega}_2, \ddot{\omega}_3)\). All the methods (inverse kinematics, 3/3/3 sensor, optical flow) have credibility, and they are related to each other. The orientation comes from the 3/3/3 sensor, which has constant credibility. The value of \(\hat{\zeta}\) and \(\hat{h}\) were defined and validated during measurements and the values can be equal to one, higher than one and lower than one. For example in ideal circumstances the credibility of all the methods would be equal. In line with the fusion of the estimated values, different special rules can be defined to handle unexpected...
circumstances like wheel scraping, magneto meter problems near electrical machines in buildings, dirty optical flow sensors, etc. [20, 21].

During the credibility based fusion of the estimated values another problem came from the different time constants of the 3/3/3 sensor, the optical flow feedback loops and the motion control loop. The optical flow sensors and the 3/3/3 sensors data registers are read at 50 Hz sampling rate but the kinematics runs at 1 kHz. According to the different time constants 3/3/3 sensor and optical flow method have recursive filters to avoid sudden transitions in the real-time position feedback loop. From the 50 Hz cycle time of the 3/3/3 sensor the time constant of the filter is 20 ms, which is acceptable according to the mechanical time constant of the approximately 40 kg robot. The estimated parameters of the robot odometry $(\hat{x}, \hat{y}, \hat{\phi})$ can be expressed as (4.1–4.3).

\[
\begin{align*}
\hat{x} &= \frac{\dot{x}_p \hat{\xi} + \dot{x}_p \hat{h}}{\xi + \hat{h}} \\
\hat{y} &= \frac{\dot{y}_p \hat{\xi} + \dot{y}_p \hat{h}}{\xi + \hat{h}} \\
\hat{\phi} &= \frac{\dot{\phi}_p \hat{\eta} + \dot{\phi}_p \hat{\rho} \cdot \left(1 + \dot{\phi}_o \hat{h}\right)}{\xi + 1 + \hat{h}}
\end{align*}
\] (30–32)

The credibility of the robot odometry $(\hat{\xi})$ can be tuned with experimental results and can be dependent from different motion related parameters as equation (33).

\[
\hat{\xi}(k_1 \frac{\Delta t_{cont}}{\Delta t_1^2 + \Delta t_2^2 + \Delta t_3^2}, k_2 \frac{\Delta t_{cont}}{\hat{\varphi}_1 + \hat{\varphi}_2 + \hat{\varphi}_3}, \ldots)
\] (33)

where $k_1$ and $k_2$ are constant parameters to optimize the function, $\Delta t_{cont}$ is the motor current, $\hat{\varphi}_1$ are the angular accelerations of the wheels and $\Delta t_{cont}$ is the motion control time period. With a fine tuning these methods could filter the unexpected dynamical events like wheel slipping. During the experiments, the odometry credibility was resulted from the change of the motor currents. The credibility of the robot odometry sensor ($\hat{h}$) can be expressed as equation (34).

\[
\hat{h} = k \frac{1}{h_1^2 + h_2^2}, \frac{S_{QUAL}}{40}
\] (34)

where $k$ is constant to optimize the exponential function, $S_{QUAL}$ is the surface register of the sensor, 40 is the reference value of the $S_{QUAL}$ register in case of white paper, $h_1$ and $h_2$ are the distances between the sensors and the ground plane (surface). According to the sensor datasheet, it measures from 3.4 mm without any problems but during the experiments we could use the sensors even around 10 mm distance. For the experimental results $k = 200$ where used. For example in case of a 10 mm distance at each sensors and the $S_{QUAL}$ register is 100 and the odometry measurement still works so $\hat{h} = 1$. Over 10 mm and $S_{QUAL} < 40$ the result can be $\hat{h} \ll 1$, under 10 mm and $S_{QUAL} = 40$ the result can be $\hat{h} = 1$.

The real-time motion controller contains a path planner, an interpolation between the position and the angular position and three PD controllers for the wheels. During the robot motion the following constant parameters can be tuned:

1. maximal velocity and angular velocity of the robot
2. maximal acceleration and angular acceleration of the robot
3. maximal angular velocity of the wheels
4. maximal angular acceleration of the wheels
5. $P$ and $D$ parameters of the wheel position controllers [22]

5 OPEN LOOP TEST

To prove the concept several open loop tests were made on the robot. In this case the embedded system measured the robot position and orientation during motion on a reference path. The grey path lines on the graphs are measured by the embedded system, the points are the initial and the final points of the path which are measured by a motion capture system and the dotted black line is the path, which was performed for the motion (reference).

The most accurate measurement (initial point and final point) of the robot position was made with the existing motion capture system of the Mechatronics, Optics and Mechanical Engineering Informatics Department of Budapest University of Technology and Economics. [23] The motion capture system could not calculate real-time the motion of the robot with the requested resolutions so we could use only the initial and the last points of the motion.

For the first time the orientation of the robot did not change during the motion Fig. 13. $(\Delta \phi_{robot}(t) = 0)$. In this case the position error was only around 20 millimeters at the end of the path Fig. 14.

For the second time the motion and the robot orientation were always parallel with the direction of the motion Fig. 15. $(\Delta \phi_{robot}(t) \neq 0)$. In this case the error was around 220 millimeters at the end of the path Fig. 16.

The distance between the real end position and the expected end position of the robot showed the same results as the last values of the error plots, which can be compensated during path planning [24-26].
6 CONCLUSION

During motion, the kiwi drive in Ethon continuously accumulated the position and the orientation errors. This paper described an implemented solution for kiwi drive position and orientation problem. The proposed method is based on the inverse kinematics, a sensor fusion between magnetometer, accelerometer, and gyroscope and on a mouse sensors based optical flow system. The mouse sensor measurements and the open loop test proved that the concept can improve the navigation of a kiwi drive based holonomic mobile robot. Refocusing the optics of the mouse sensor based system the accuracy can be further improved. The increasing market and needs of the service robot sector require easy-to-implement and low cost solutions in mobile robotics. By the embedded system based service robots with limited calculation throughput the indoor localization is an ongoing problem.

ACKNOWLEDGEMENT

The authors wish to thank the support to the Hungarian Research Fund (OTKA K100951). This research was funded by the Hungarian Academy of Sciences (MTA 01 031).
REFERENCES


Optical flow based odometry for mobile robots supported by multiple sensors and sensor fusion

F. Tajti, G. Szayer, B. Kovács, P. Barna, P. Korondi

AUTHORS’ ADDRESSES

F. Tajti received MSc. in 2012 at Mechatronics at Budapest University of Technology and Economics. Since then he is Ph.D. student at the Department of Mechatronics Optics and Mechanical Engineering Informatics in the Faculty of Mechanical Engineering. During his engineering school he got several Scientific Student Conference awards. Since 2012 he is working in the MTA-ELTE Comparative Ethological Research Group (MTA 01 031) of the Hungarian Academy of Sciences. During his whole professional carrier he worked on or even coordinated several engineering project in the field of industrial robot control and mobile robotics. He developed for the Narvik University College in Norway, for the Institute for Computer Science and Control of the Hungarian Academy of Sciences.

G. Szayer received MSc. in 2011 at Mechatronics at Budapest University of Technology and Economics (BME). During his engineering school he got several Scientific Student Conference awards: two first-, and three second- and more other prices. He worked 1.5 years in the Budapest R&D Center of Knorr-Bremse as a function development embedded software engineer before applying to the PhD course of the Department of Mechatronics Optics and Engineering Informatics at the Faculty of Mechanical Engineering of BME. He founded his own company – General Mechatronics Ltd. – in 2012 with Bence Kovacs, and works on different industrial and R&D projects in the field of industrial- and mobile robotics.

B. Kovács received MSc. degree in 2011 at Mechatronics at Budapest University of Technology and Economics. During his engineering school he got several Scientific Student Conference awards: two first-, and three second- and more other prices. Before applying to PhD course of the Department of Mechatronics Optics and Engineering Informatics at the Faculty of Mechanical Engineering of BME, he worked two years in the Budapest R&D Center of Ericsson. Within this time, he worked as software developer then worked as hardware developer of high speed network systems. He founded his own company – General Mechatronics Ltd. – in 2012 with Géza Szayer, and works on different industrial and R&D projects in the field of industrial- and mobile robotics.

P. Barna received BSc in 2013 at Mechatronics at Budapest University of Technology and Economics, and is now an MSc student. There he became interested and involved in the theoretical background of mobile robotics. He also plans to be a PhD student in the field of mobile robotics related computer vision. During the last two semesters he worked on project Ethon at the Department of Mechatronics, Optics, and Engineering Informatics Budapest University of Technology and Economics especially on the theoretical part and the implement of the optical flow method. He also works for a small firm specialised in machine vision solutions.

P. Korondi (M’98-SM’11) received the Dipl. Eng. and Ph.D. degrees in electrical engineering from the Technical University of Budapest, Budapest, Hungary, in 1984 and 1995, respectively. Since 1986, he has been with Budapest University of Technology and Economics (formerly the Technical University of Budapest). From April 1993 to April 1995, he worked in the laboratory of Prof. Harashima and Prof. Hashimoto at the Institute of Industrial Science, The University of Tokyo, Japan, where he continues to spend a month each year, working on joint research. As a result of this cooperation, the Intelligent Integrated System Japanese–Hungarian Joint Laboratory was founded in 2001. His research interests include telemanipulation and motion control. Dr. Korondi is a founding member of the International PEMC Council. He is also with the MTA-BME Control Engineering Research Group of the Hungarian Academy of Sciences.

Received: 2014-06-03
Accepted: 2015-05-23