Development of a Three-Dimensional Ball Rotation Sensing System using Optical Mouse Sensors

Masaaki Kumagai and Ralph L. Hollis

***Abstract*— Robots using ball(s) as spherical wheels have the advantage of omnidirectional motion. Some of these robots use only a single ball as their wheel and dynamically balance on it. However, driving a ball wheel is not straightforward. These robots usually use one or more motor-driven rollers or wheels in frictional contact with the ball wheel. Some slippage can occur with these schemes, so it is important to measure the actual rotation of the ball wheel for good control. This paper proposes one method for measuring the three dimensional rotation of a sphere, which is applicable to the case of a ball wheel. The system measures surface speed by using two or more optical mouse sensors and transforms them into the angular velocity vector of the ball, followed by integration to give the rotational angle. Experiments showed the correctness of this method, yielding an error of approximately 1%, with a 10 ms response.**

I. Introduction

Robots that dynamically balance using a single ball as a spherical wheel are being increasingly developed. The first successful robot was the *ballbot* [1], [2], followed by *BallIP* [3], [4] with a different type of ball drive, and many subsequent robots were developed with similar ball drive mechanisms. Whereas there are several types of ball drives, one common feature of these ball balancing robots are that they drive the ball by friction. Some have motor-driven rollers and others have wheels or crawlers, whose motion is transferred to the ball wheel at contact points between them. At each contact point, the ball usually slips a little because the motion of one roller (wheel) is prevented by other rollers. There may be a large slip if an excessive force/torque is applied. These slips will cause problems of mismatch between the calculated ball rotation and the actual rotation, resulting in failure of the feedback control in the worst case. Therefore, the direct measurement of ball rotation, as opposed to measuring the motion of the drive rollers (wheels) is highly desirable in practical applications of these robots. This paper describes ball angular motion sensing for such cases.

The proposed method uses more than one optical (laser) mouse sensors. The idea to use mouse sensors for robot motion sensing is not new and there are several previous works. Of course, for two-dimensional sensing, optical track balls are commercially available.

The work most similar to this paper is that proposed by K. M. Lee and D. Zhou [5]. The authors used a pair of mouse sensors in parallel. Their method makes the sensing module small, but the equations needed to calculate the rotation are complex and the expression of the output angles is not easy to use for robot control. Compared to their work, the method proposed below allows an almost free arrangement of sensors, improvement of signal-to-noise ratio by averaging multiple outputs, and simple calculations and outputs.

There are several reports of using mouse sensors as a odometry sensors for a planar mobile robot [6], [7]. The sensors were attached directly to the bottom of the wheeled robot. Although the present paper mainly concerns sensing on a spherical surface, we also describe an extension for planar sensing by assuming the ball radius goes to infinity. Another interesting work [8] uses an optical sensor for assessing the surface condition. The surface condition of the ball is also important for the balancing robot, and their idea would be useful in combination with ball rotation sensing.

In summary, this paper describes a method for sensing a ball angular velocity using optical mouse sensors. We also present a method for averaging multiple measurements from more than two sensors, and a method for integrating angular velocity to determine the ball rotation angle. The method was evaluated using an experimental setup with three optical mouse sensors, and it could measure complex motion of the ball within 1% error.

II. Sensing Method

A. Optical Sensing for a Mouse

The well-known sensor for the optical mouse is a set of imaging devices coupled to an image processor [5], [9]. The imaging device has a small number of pixels. The processor detects the motion of the image acquired by the device, *i.e.*, motion of the target and/or that of the sensor. Cheaper optical mice use LEDs as their light sources and can detect slower motion with lower resolution. More expensive mice, called "laser mice" or "gaming mice," have laser diodes as the light sources and can detect much more precise motion with higher speed, for example, up to 3.8 m/s (150 ips) in speed and 5 μm (5000 cpi) resolution with a processing rate of 12000 fps.

Such optical sensors usually sense *relative displacement* along two axes in one processing frame period and accumulate the displacement. The mouse for the PC has a microcontroller, and it reads the displacement from the sensor in a specific cycle and sends the value via a USB (or PS/2) interface. Because the controller reads and reports at a constant rate, the results can be assumed to be the velocities along the pair of orthogonal axes. The sensing device is affected by optical conditions because it is a kind

Masaaki Kumagai is with the Faculty of Engineering, Tohoku Gakuin University, Tagajo 985-8537 Japan. Ralph L. Hollis is with The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA.
kumagai@tjcc.tohoku-gakuin.ac.jp,
rhollis@cs.cmu.edu

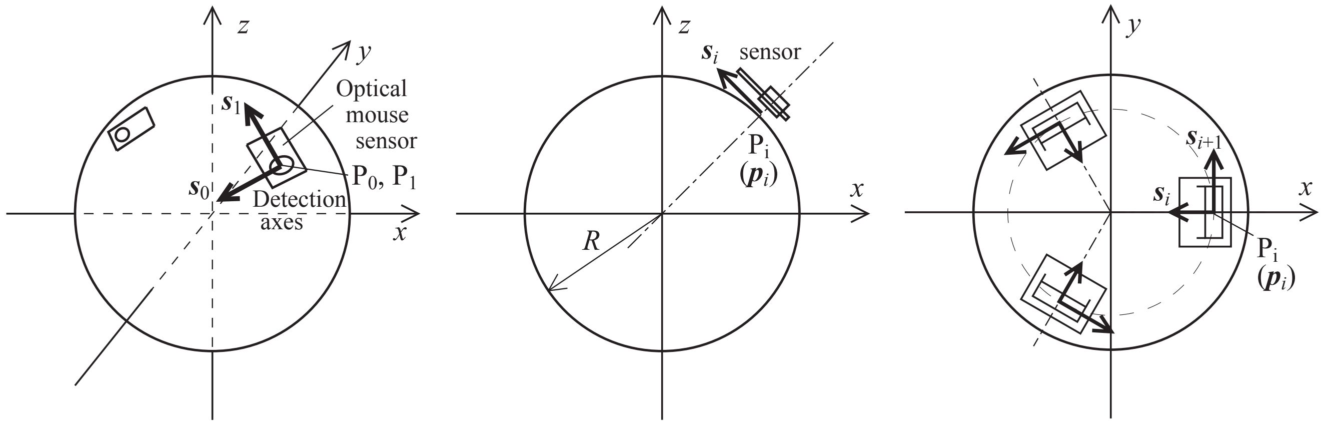


Fig. 1. Example arrangements of mouse sensors around a ball.

of camera. The distance between the sensor and the surface is an important factor because not only the image focus but also the magnification depends on it. The surface pattern of the target object (usually a mouse pad or desktop) is also important for the image matching process, *i.e.*, a glossy surface has no significant features but matte or scratched surfaces are easily tracked. These optical influences must be considered in practical implementations.

B. Sensing of the Ball Angular Velocity

To sense the angular velocity vector of a ball, a set of three surface velocities measured by the mouse sensors is converted as follows. Because a common mouse sensor has two orthogonal sensing axes, at least two sensors are needed for the three velocities. We assume there are N mouse sensors that provide $n = 2N$ velocity readings. Sensor l, ($l = 0 \dots N-1$) has two velocity measurements of $2l$ and $2l+1$. The sensors are arranged on the surface of the ball as shown in Fig. 1. Each sensor P$_i$ ($i = 0 \dots n-1 (= 2N-1)$), has a position vector $\boldsymbol{p}_i$, and a unit vector along its sensing axis $\boldsymbol{s}_i$ which are assumed to be known. Note that in most mouse sensors, $\boldsymbol{p}_{2l} = \boldsymbol{p}_{2l+1}$, $\boldsymbol{s}_{2l} \cdot \boldsymbol{s}_{2l+1} = 0$, and $\boldsymbol{s}$ is perpendicular to the normal at the point.

First, three sensed velocities, v_{si}, v_{sj}, v_{sk} are chosen among n sensor readings ($i, j, k = 0 \dots n-1$). These three velocities are of course not the same readings ($i \neq j, j \neq k, k \neq i$), but two of them may be a pair of readings from one mouse sensor (*e.g.* $\boldsymbol{p}_i = \boldsymbol{p}_j$). They also must satisfy the following additional condition.

When the ball is rotating with an angular velocity vector of $\boldsymbol{\omega}$, the circumferential speed $\boldsymbol{v}_i$ of the ball at P$_i$ is:

$$\boldsymbol{v}_i = \boldsymbol{\omega} \times \boldsymbol{p}_i. \tag{1}$$

The component of $\boldsymbol{v}_i$ along the sensing direction $\boldsymbol{s}_i$ for sensor i can be obtained from the inner product

$$\begin{aligned} v_{si} &= \boldsymbol{s}_i \cdot \boldsymbol{v}_i = \boldsymbol{s}_i \cdot (\boldsymbol{\omega} \times \boldsymbol{p}_i), \\ &= \boldsymbol{\omega} \cdot (\boldsymbol{p}_i \times \boldsymbol{s}_i). \end{aligned} \tag{2}$$

With the same equations for sensing j and k, they can be written in matrix form as

$$\begin{pmatrix} v_{si} \\ v_{sj} \\ v_{sk} \end{pmatrix} = \begin{pmatrix} \boldsymbol{p}_i \times \boldsymbol{s}_i \\ \boldsymbol{p}_j \times \boldsymbol{s}_j \\ \boldsymbol{p}_k \times \boldsymbol{s}_k \end{pmatrix} \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} = \boldsymbol{A}\boldsymbol{\omega}. \tag{3}$$

$\boldsymbol{A}(i, j, k)$ is similar to the Jacobian matrix between the ball angular velocity and the three circumferential speeds.

When the determinant of $\boldsymbol{A}$ is not zero, this equation can be solved and $\boldsymbol{\omega}$ is calculated from the velocity triplet $(v_{si}, v_{sj}, v_{sk})^T$:

$$\boldsymbol{\omega}(i, j, k) = \boldsymbol{A}^{-1} \begin{pmatrix} v_{si} \\ v_{sj} \\ v_{sk} \end{pmatrix}, \tag{4}$$

which is the fundamental equation of this method. The angular velocity of the ball can be obtained by measuring three circumferential speed components using three velocity sensors. The requirement is that the triplet of $\boldsymbol{p}_i \times \boldsymbol{s}_i$ must be chosen so that they are linearly independent, *i.e.*, $|\boldsymbol{A}| \neq 0$. For instance, if $\boldsymbol{p}_i = -\boldsymbol{p}_j$ and $\boldsymbol{s}_i = \pm\boldsymbol{s}_j$, *i.e.*, two sensors are arranged symmetrically about the ball center, $|\boldsymbol{A}|$ becomes zero.

C. Using Multiple Triplets of Sensors

In the system with N mouse optical sensors, one can obtain typically $n = 2N$ velocity readings as mentioned above. One can choose $C(n, 3) = n(n-1)(n-2)/6$ triplets from these readings. Therefore, a weighted average of the angular velocity vector from each triplet can be calculated.

First, some of the triplets can be rejected by the condition $|\boldsymbol{A}| \neq 0$ based on the layout of $\boldsymbol{p}_i$ and $\boldsymbol{s}_i$. The triplets with relatively smaller $|\boldsymbol{A}|$ are also negligible because their $\boldsymbol{A}^{-1}$ tend to be larger, which will magnify velocity errors as mentioned below. The sensor used in our experiments has a ±1 bit fluctuation in resolution similar to other digital type sensors with small statistical errors.

Two scalar values of each triplet are used as weights to help compensate any error trends of the sensor. One of the weights is $w_1(i, j, k) = ||\boldsymbol{A}||$ (an absolute value of the determinant). The error will be more significant if elements of $\boldsymbol{A}^{-1}$ are larger, which is associated with a smaller

$||\boldsymbol{A}||$. This value is a figure of merit for the geometrical arrangement of a set of three sensors, which depends only on mechanical design and the selection of each triplet.

The other weight is based on the sign of the elements of $\boldsymbol{A}^{-1}$. Equation (4) calculates each angular velocity as

$$\omega_{x(or\ y,z)} = a_1 v_{si} + a_2 v_{sj} + a_3 v_{sk}, \tag{5}$$

where a_1, a_2, a_3 are corresponding elements of $\boldsymbol{A}^{-1}$.

The second weight is defined as:

$$w_{2x(y,z)}(i,j,k) = \frac{|a_1 v_{si} + a_2 v_{sj} + a_3 v_{sk}|}{|a_1 v_{si}| + |a_2 v_{sj}| + |a_3 v_{sk}|}. \tag{6}$$

This positive value is unity if all terms have same sign, and it is less than 1 if there are both positive and negative terms in av_s. This term is introduced because the number of significant digits is usually decreased by subtracting values. A symmetrically arranged set of sensors sometimes results in a calculation of 0 when subtracting a large value from another large value, which can be easily affected by the sensing errors of the raw velocities. The weight w_2 becomes smaller in such cases.

Using these two types of weights, the final result is obtained by averaging $\boldsymbol{\omega}$ over all triplets:

$$\overline{\omega_x} = \frac{\sum_{i,j,k} w_1(i,j,k) w_{2x}(i,j,k)\ \omega_x(i,j,k)}{\sum_{i,j,k} w_1(i,j,k) w_{2x}(i,j,k)}, \tag{7}$$

with $\overline{\omega_y}$ and $\overline{\omega_z}$ calculated in the same manner.

D. Calculation of the Ball Rotation

The angular velocity vector of the ball is obtained from the sensor readings as discussed above. There are many expressions for three-dimensional rotation, *e.g.*, Euler angles, but here we use a bare rotation matrix. Its three columns express the unit vectors of a Cartesian coordinate frame fixed at the center of the ball. Most of the angle expressions can be derived from the rotation matrix, and many practical applications require the transform matrix instead of the Euler angles. The objective of this work, sensing of the ball wheel rotation for ball-wheel robots, also does not require the Euler angles.

We assume that the calculation of the ball rotation $\overline{\boldsymbol{\omega}}$ is performed periodically with an interval of T_s.

The rotation matrix $\boldsymbol{R}$ is updated in each calculation cycle.

$$\begin{aligned} \boldsymbol{R}_{new} \quad = & \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\Delta\theta_x & -\sin\Delta\theta_x \\ 0 & \sin\Delta\theta_x & \cos\Delta\theta_x \end{pmatrix} \\ & \begin{pmatrix} \cos\Delta\theta_y & 0 & \sin\Delta\theta_y \\ 0 & 1 & 0 \\ -\sin\Delta\theta_y & 0 & \cos\Delta\theta_y \end{pmatrix} \\ & \begin{pmatrix} \cos\Delta\theta_z & -\sin\Delta\theta_z & 0 \\ \sin\Delta\theta_z & \cos\Delta\theta_z & 0 \\ 0 & 0 & 1 \end{pmatrix} \boldsymbol{R}_{old}, \end{aligned} \tag{8}$$

where $\Delta\theta_x = T_s\overline{\omega_x}$, $\Delta\theta_y = T_s\overline{\omega_y}$, $\Delta\theta_z = T_s\overline{\omega_z}$. The order of each rotation around x, y, and z axes is usually important, but $\Delta\theta_x$, $\Delta\theta_y$, and $\Delta\theta_z$ are small enough when the period T_s is short, which makes the order much less significant. We have not examined the effect of changing T_s, although we know a longer T_s will cause numerical errors in integration, whereas a shorter T_s may have relatively larger quantizing error of the velocity readings. The columns of the matrix, *i.e.*, the unit vectors fixed on the ball, are used for evaluating the measurements in the following experiments.

E. Sensing for Planar Motion

The idea of sensing ball rotation can also be applied to sensing planar motion, by assuming the radius R of the ball goes to infinity. For instance, when the motion is assumed to be performed around the top of the ball, translational velocities $v_x = R\omega_y$ and $v_y = -R\omega_x$, with ω_z as the angular velocity normal to the plane while ω_x and ω_y approach 0. Because this is the limiting condition, the equations must be modified.

Equation (3) is expanded first.

$$\begin{aligned} & \begin{pmatrix} v_{si} \\ v_{sj} \\ v_{sk} \end{pmatrix} = \begin{pmatrix} \boldsymbol{p}_i \times \boldsymbol{s}_i \\ \boldsymbol{p}_j \times \boldsymbol{s}_j \\ \boldsymbol{p}_k \times \boldsymbol{s}_k \end{pmatrix} \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} = \\ & \begin{pmatrix} p_{iy}s_{iz} - p_{iz}s_{iy} & p_{iz}s_{ix} - p_{ix}s_{iz} & p_{ix}s_{iy} - p_{iy}s_{ix} \\ p_{jy}s_{jz} - p_{jz}s_{jy} & p_{jz}s_{jx} - p_{jx}s_{jz} & p_{jx}s_{jy} - p_{jy}s_{jx} \\ p_{ky}s_{kz} - p_{kz}s_{ky} & p_{kz}s_{kx} - p_{kx}s_{kz} & p_{kx}s_{ky} - p_{ky}s_{kx} \end{pmatrix} \\ & \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix}. \end{aligned} \tag{9}$$

For the planar sensing case, one can assume $p_{i(j,k)z} = R$ and $s_{i(j,k)z} = 0$ without losing generality, which means the sensors are located at the top surface of an extremely large ball with radius of R ($R \to \infty$) and the sensing directions of the sensors are parallel to the (horizontal) tangential surface. This gives

$$\begin{aligned} & \begin{pmatrix} v_{si} \\ v_{sj} \\ v_{sk} \end{pmatrix} \\ = & \begin{pmatrix} -Rs_{iy} & Rs_{ix} & p_{ix}s_{iy} - p_{iy}s_{ix} \\ -Rs_{jy} & Rs_{jx} & p_{jx}s_{jy} - p_{jy}s_{jx} \\ -Rs_{ky} & Rs_{kx} & p_{kx}s_{ky} - p_{ky}s_{kx} \end{pmatrix} \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} \\ = & \begin{pmatrix} s_{iy}(-R\omega_x) & +s_{ix}R\omega_y & +(p_{ix}s_{iy} - p_{iy}s_{ix})\omega_z \\ s_{jy}(-R\omega_x) & +s_{jx}R\omega_y & +(p_{jx}s_{jy} - p_{jy}s_{jx})\omega_z \\ s_{ky}(-R\omega_x) & +s_{kx}R\omega_y & +(p_{kx}s_{ky} - p_{ky}s_{kx})\omega_z \end{pmatrix} \\ = & \begin{pmatrix} s_{iy}v_y & +s_{ix}v_x & +(p_{ix}s_{iy} - p_{iy}s_{ix})\omega_z \\ s_{jy}v_y & +s_{jx}v_x & +(p_{jx}s_{jy} - p_{jy}s_{jx})\omega_z \\ s_{ky}v_y & +s_{kx}v_x & +(p_{kx}s_{ky} - p_{ky}s_{kx})\omega_z \end{pmatrix} \\ = & \begin{pmatrix} s_{ix} & s_{iy} & p_{ix}s_{iy} - p_{iy}s_{ix} \\ s_{jx} & s_{jy} & p_{jx}s_{jy} - p_{jy}s_{jx} \\ s_{kx} & s_{ky} & p_{kx}s_{ky} - p_{ky}s_{kx} \end{pmatrix} \begin{pmatrix} v_x \\ v_y \\ \omega_z \end{pmatrix} \\ & = \widetilde{\boldsymbol{A}} \begin{pmatrix} v_x \\ v_y \\ \omega_z \end{pmatrix}. \end{aligned} \tag{10}$$

In the above derivation, $-R\omega_x = v_y$ and $R\omega_y = v_x$ are introduced, and the order of the terms concerning v_x and v_y

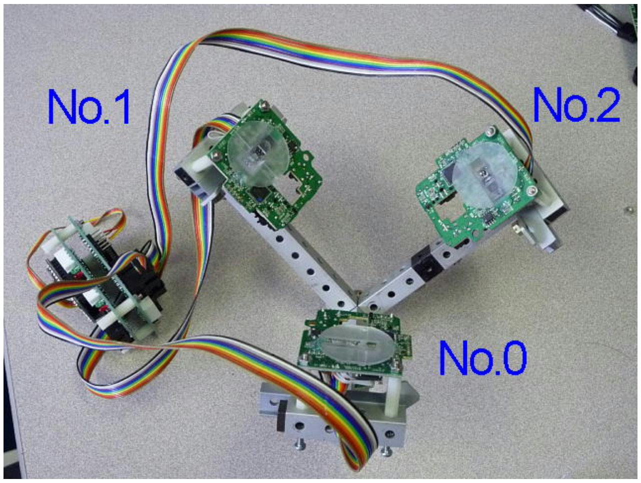


Fig. 2. Three mouse sensor boards with fixing frame.

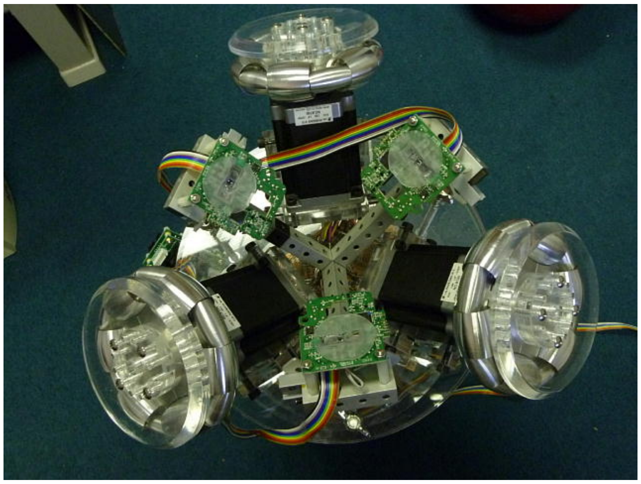


Fig. 3. Sensor frame mounted on *BallIP* drive.

are exchanged. Thus one obtains a result similar to that of Eq. (3). The process following Eq. (3) can be applied to this planar model to obtain translational velocities and rotational angular velocity on the flat surface using two or more mouse sensors.

III. Experiments

To evaluate the validity of the idea and its accuracy, both a ball motion sensing setup and a planar motion sensing prototype were developed. The system for the ball is described below and the one for planar sensing is demonstrated in the accompanying video.

A. Experimental setup

The experimental setup for sensing ball motion consists of three components.

- A sensing frame with three laser mouse sensors (ADNS-6010, Avago technologies) and one controlling microprocessor.
- A Windows PC for calculating angular velocity and rotation angle of the ball.
- A ball drive mechanism to generate specific repeatable motions of the ball.

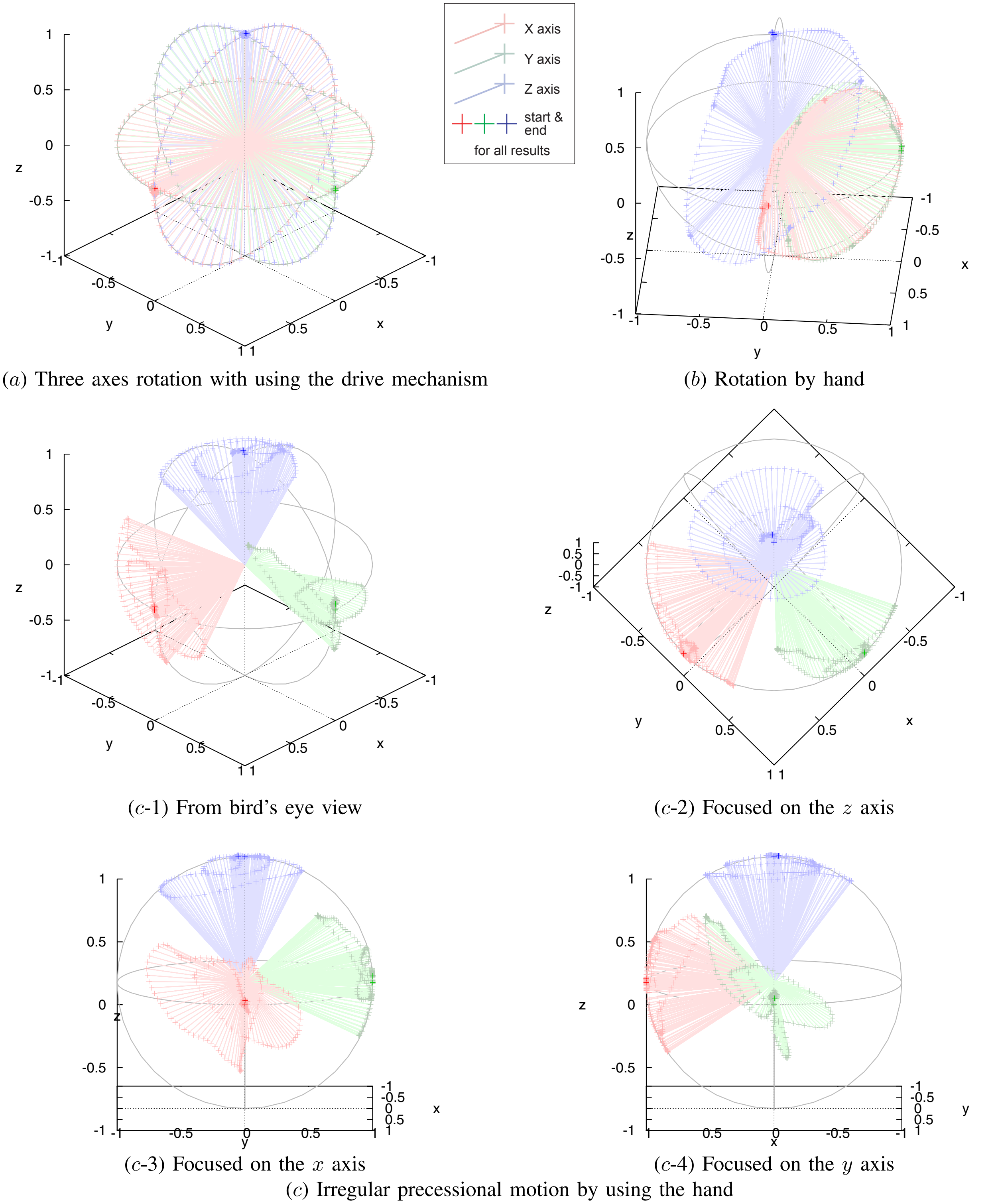
Figure 2 shows three mouse sensors fixed on a frame. The frame was designed to fit the ball drive of the *BallIP* balancing robot [3] as in Fig. 3. Three sensors were fixed at an elevation of 54.4°, in equal spacing of 120° in the horizontal plane with one sensing axis of each sensor on the horizontal plane as shown in the right side of Fig. 1. One of the sensors is located above the x axis. Note that the z axis of the system goes down in Fig. 3, whereas it is directed upward in the following results because the ball drive in the picture is inverted from its intended use in a balancing robot.

The ADNS-6010 sensor was chosen for the system because of its availability. It is used in several popular mice such as the “Microsoft SideWinder X5 Mouse.” There is an evaluation kit for the sensor, but we used the circuit board of that mouse because it was much cheaper than the kit and it also includes a suitable laser diode and an assembled optical system. A special controller was developed and used instead of using the original USB mouse controller for two reasons. One reason was the difficulty in obtaining raw output from the mouse. In the first test setup, a Linux PC and a software library *libusb* was used, but connecting three mice to the PC and probing the correspondence between physical mice and their logical identifications was bothersome. The second reason was to enable near-simultaneous sensing from the three mice. The mice connected to the USB port report their motion data within the same period (125 Hz for an ordinary mouse and 500 Hz for many of the gaming mice), but the timing of the sensor (read from the sensing device) and the transmission via USB differ from each other.

The developed controller for the ADNS-6010 using the dsPIC microprocessor (Microchip) can handle up to four ADNS-6010s and can read data simultaneously from them within a 1 kHz cycle. It can report the velocity results up to a 1 kHz rate using a 115,200 bps serial port (the report will be at a 250 Hz rate if four sensors are used). The controller can also download firmware necessary for the ADNS sensor.

The PC receives data from the serial port (a serial-USB converter was used), and calculates the angular velocity vector and the rotation of the ball as described above. The software also displayed the data and recorded it for later analysis. A planar sensing system was also tested, which is demonstrated in the video. The major difference between the two setups was the shape of the frame. Four sensors were mounted on a frame with legs on both sides so that sensors floated 1 mm above the sensing surface (a desktop in the video).

The last component is the ball driving mechanism. To carry out repeatable experiments, the ball drive of the *BallIP* robot was used. Special software was installed on the controller of the robot instead of using its normal balancing control, so it could rotate the ball in multiples of 360° around the x, y, and z axes on command. Each of rotating motions was calibrated with an error of less than 1° for each 360° revolution.



(a) Three axes rotation with using the drive mechanism

(b) Rotation by hand

(c-1) From bird's eye view

(c-2) Focused on the z axis

(c-3) Focused on the x axis

(c-4) Focused on the y axis

(c) Irregular precessional motion by using the hand

Fig. 4. Experimental results of ball motion sensing. Points with lines indicate trajectories of unit vectors fixed at the center of the ball.

TABLE I

END POINTS OF THE TRAJECTORIES AND SENSING ERROR IN DEGREES.

Motion	Unit vectors [error in degree]		
Expected result (Initial point)	(1.000, 0.000, 0.000) [0.0 deg]	(0.000, 1.000, 0.000) [0.0 deg]	(0.000, 0.000, 1.000) [0.0 deg]
Rotation around three axes	(1.000, 0.006, 0.020) [1.2 deg]	(-0.006, 1.000,-0.002) [0.4 deg]	(-0.020, 0.002, 1.000) [1.2 deg]
Rotation by the hand	(0.999, 0.043, 0.028) [2.9 deg]	(-0.043, 0.999, 0.017) [2.6 deg]	(-0.028,-0.017, 1.000) [1.9 deg]
Irregular precession	(0.999, 0.003, 0.035) [2.0 deg]	(-0.004, 0.999, 0.052) [3.0 deg]	(-0.035,-0.052, 0.998) [3.6 deg]

The ball used as a target was taken from the *ballbot* balancing robot [1], which is approximately 200 mm in diameter and coated with urethane. The ball was additionally coated with matte varnish so that it did not reflect the laser like a mirror. According to the data sheet, the ADNS-6010 has a resolution of 2000 cpi, 0.0127 mm, a frame rate of up to 7080 fps, and a maximum speed of up to 45 ips, 1.1 m/s. With this ball diameter, the resolution becomes 0.007° and the maximum angular velocity is 1.8 rev/s, or 650°/s.

B. Experimental results

Typical results are shown in Fig. 4. The results showed the trajectories of the three unit vectors of the coordinate frame fixed on the ball, which are components of the rotation matrix in Eq. (8). The vectors started from $(1, 0, 0)$ (red), $(0, 1, 0)$ (green), and $(0, 0, 1)$ (blue). The points in the results are plotted at 40 ms intervals (25 Hz) whereas the calculation was done at 100 Hz. In each case, the motion of the ball ends up at almost the same posture as it had at the start, ensured by markers on the ball and a laser fiducial line. The position error of the markers on the ball was less than 1 mm, which is less than 0.5° rotational error of the 200 mm diameter ball. As shown in Fig. 4, each of the axis trajectories in all of the results has two distinguishable bright points indicating the start and end points of the motion, which should coincide if there is no error.

Figure 4(a) shows the case of the ball being rotated 360° around the x, y, and z axes sequentially using the *BallIP* drive operating at a speed of approximately 100°/s. Therefore, the trajectory of the axes track the ball equator and two meridians on the $x - y$, $y - z$ and $z - x$ planes. Figure 4(b) shows one revolution of the ball performed by hand. The axis of the rotation is approximately $(1, 1, 0)$ but was not constant. The set of results shown in Fig. 4(c) was taken while the ball was rotated by hand using a precession motion. In this case, the motion of the ball was irregular and the error caused by the accumulation of angular velocity was larger. These motions are demonstrated in the accompanying video (the data was taken simultaneously with the video).

Table I shows numerical results for the three motions discussed above. It shows the x, y and z components of the three unit vectors (denoted with points and lines in the figures) in the final states of each motion. These vectors were expected to be the same as those of the initial states, but contained errors (shown as small distances between two bright points in the figures). The table also contains these errors in units of degrees. All errors are almost within 3.6°, which is equivalent to 1% error in each revolution. These results indicate that the angular velocity sensing also has an accuracy of 1%, on average.

As is the nature of velocity-type sensing schemes, the error in the position (angle) increases gradually whereas it has good precision and response speed in angular velocity sensing and short-time relative motion sensing. For long term use or an application that requires the absolute posture indefinitely, this error must be compensated by an absolute sensing method such using a patterned ball [10], which may be a much slower method.

IV. Conclusions

A sensing method for the angular velocity vector and the rotational motion of a ball was proposed in this paper. The method uses multiple optical mouse laser sensors to obtain the surface velocity of the ball, followed by simple equations to calculate its angular velocity. The position and the orientation of each sensor have few restrictions and one can freely arrange the sensors in consideration of other components of the system. Two types of weighting functions were introduced for averaging multiple sets of sensor outputs. The paper emphasizes using triplets of mouse sensors whose accuracy depends on their positions and orientations around the ball. The method is extended to planar sensing by assuming that the radius of the ball goes to infinity. An experimental setup for evaluating the method was built using the *BallIP* robot drive. The setup had a precision of less than 1% error in its rotation ability. Although the system was developed for a robot balancing on its ball, we project there will be other important uses.

V. Acknowledgements

This work was performed in the Microdynamic Systems Laboratory, The Robotics Institute, Carnegie Mellon University, as a part of the dynamically stable mobile robots project. Masaaki Kumagai was supported by a sabbatical system of Tohoku Gakuin University.

References

[1] T.B.Lauwers, G.A.Kantor, R.L.Hollis, “A Dynamically Stable Single-Wheeled Mobile Robot with Inverse Mouse-Ball Drive,” *Proc. ICRA 2006*, 2006, pp 2884–2889

[2] U.Nagarajan, M.Anish Mampetta, G.Kantor, R.Hollis, “State Transition, Balancing, Station Keeping, and Yaw Control for a Dynamically Stable Single Spherical Wheel Mobile Robot,” *Proc. ICRA 2009*, 2009, pp 998–1003

[3] M.Kumagai, T.Ochiai, “Development of a robot balanced on a ball – Application of passive motion to transport –,” *Proc. ICRA 2009*, 2009, pp 4106–4111

[4] M.Kumagai, T.Ochiai, “Development of a Robot Balanced on a Ball – First Report, Implementation of the Robot and Basic Control –,” *Journal of Robotics and Mechatronics*, vol.22 no.3, 2010, pp 348–355

[5] K.M.Lee, D.Zhou, “A real-time optical sensor for simultaneous measurement of three-DOF motions,” *IEEE/ASME Trans. on Mechatronics*, vol.9 issue.3, 2004, pp 499-507

[6] D.Sekimori, F.Miyazaki, “Self-localization for indoor mobile robots based on optical mouse sensor values and simple global camera information,” *Proc. ROBIO 2005*, pp 605–610

[7] P.L.Wu, S.L.Jeng, W.H.Chieng, “Least squares approach to odometry based on multiple optical mouse sensors,” *Proc. 5th Industrial Electronics and Applications*, 2010, pp 1573-1578

[8] W.Xin, K.Shida, “Optical mouse sensor for detecting height variation and translation of a surface,” *Proc. Industrial Technology 2008*, 2008, pp 1–6

[9] Avago technologies, “ADNS-6010 Laser Mouse Sensor Data Sheet, AV02-1410EN,” *www.avagotech.com*, 2009

[10] D.Stein, E.R.Scheinerman, G.S.Chirikjian , “Mathematical Models of Binary Spherical-Motion Encoders,” *IEEE/ASME trans. on Mechatronics*, vol.8 no.2, 2003, pp 234–244