Proceedings of ICAR 2003
The 11th International Conference on Advanced Robotics
Coimbra, Portugal, June 30 - July 3, 2003

Camera-Inertial Sensor modelling and alignment for Visual Navigation

João Alves **Jorge Lobo** **Jorge Dias**
jalves@isr.uc.pt jlobo@isr.uc.pt jorge@isr.uc.pt

Dept. of Electrical and Computer Engineering
Institute for Systems and Robotics
University of Coimbra - Polo II
3030-290 Coimbra, Portugal

Abstract

Inertial sensors attached to a camera can provide valuable data about camera pose and movement. In biological vision systems, inertial cues provided by the vestibular system, are fused with vision at an early processing stage. Vision systems in autonomous vehicles can also benefit by taking inertial cues into account.

In order to use off-the-shelf inertial sensors attached to a camera, appropriate modelling and calibration techniques are required. Camera calibration has been extensively studied, and standard techniques established. Inertial navigation systems, relying on high-end sensors, also have established techniques. This paper presents a technique for modelling and calibrating the camera integrated with low-cost inertial sensors, three gyros and three accelerometers for full 3D sensing. Using a pendulum with an encoded shaft, inertial sensor alignment, bias and scale factor can be estimated. Having both the camera and the inertial sensors observing the vertical direction at different poses, the rigid rotation between the two frames of reference can be estimated. Preliminary simulation and real data results are presented.

1 Introduction

Internal sensing using inertial sensors is very useful in mobile robotic systems and autonomous vehicles since it is not dependent on any external references, except for the gravity field which does provide an external reference. Artificial vision systems can provide better perception of the vehicle's environment by using the inertial sensor measurement of camera pose (rotation and translation). As in human vision, low level image processing should take into account the ego motion of the observer.

This paper presents a technique for modelling and calibrating the camera integrated with inertial sensors. Having both the camera and the inertial sensors observing the vertical direction at different poses, the rigid rotation between the two frames of reference shown in figure 1 can be estimated.

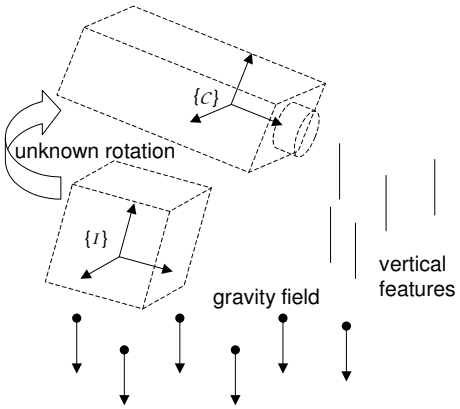


Fig. 1: *Observing gravity with the camera and the inertial sensors, the unknown rotation can be determined.*

This work is part of ongoing research into the fusion of inertial sensor data in artificial vision systems for applications on autonomous vehicles such as CyberCars [1]. In [2] and [3] the inertial sensors and system prototype are described and results presented for ground plane segmentation, in [4] a method is proposed for camera focal distance calibration using a vanishing point and the vertical reference, in [5] a vertical line segmentation method is described that performs the 3D reconstruction and mapping of the detected vertical line segments.

This paper is organized as follows: in section 2 the data from the inertial sensors is considered. A calibration method using a pendulum with an encoded shaft is presented to estimate inertial sensor alignment, bias and scale factor. Section 3 introduces the camera model and the properties of vanishing points.

The following section presents the estimation of the rigid rotation between the inertial sensors and camera frames of reference. In section 5 results are presented for both inertial sensor calibration and frame rotation estimation.

2 Data from Inertial Sensors

Inertial sensors measure linear acceleration and angular velocity. An inertial measurement unit (IMU) has three orthogonal accelerometers and three orthogonal rate gyros. To estimate velocity and position integration over time has to be performed, leading to unbounded error. The gyros keep track of rotations, so that linear velocity and position are computed in the correct frame of reference. Appropriate calibration has to be performed to minimise the error buildup.

When using inertial sensors, scale factor, bias and axis-alignment need to be known. For low cost inertial sensors these parameters are not always provided by the manufacturer, and when using discrete components their alignment has to be measured.

2.1 Intrinsic Calibration

Some of the inertial sensors parameters can be determined by performing simple operations and measuring the sensors outputs, others can not be so easily determined.

Equation (1), represents a simple model for each set of three non-coplanar accelerometers or rate gyros, which accounts for the three main errors in these sensors.

$$\begin{aligned} \boldsymbol{z}_o &= \boldsymbol{M} \cdot \boldsymbol{z}_i + \boldsymbol{b} \\ &= \begin{bmatrix} s_{xx} & s_{xy} & s_{xz} \\ s_{yx} & s_{yy} & s_{yz} \\ s_{zx} & s_{zy} & s_{zz} \end{bmatrix} \cdot \begin{bmatrix} z_{ix} \\ z_{iy} \\ z_{iz} \end{bmatrix} + \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} \end{aligned} \quad (1)$$

The quantities to be measured are represented by the vector $\boldsymbol{z}_i$, while $\boldsymbol{z}_o$ represents the actual output from the sensors. Vector $\boldsymbol{b}$ represents the bias for each individual sensor, while s_{kk} is the sensitivity (or scale factor) for the sensor oriented along axis k, and s_{kl} the cross sensitivity, resulting from axis misalignments, relating axis k and l.

2.2 Calibration with a Pendulum

In this work, a pendulum is used in order to determine the inertial sensors' parameters. The pendulum was chosen since it is relatively straightforward to determine the real quantities the sensors are measuring.

To get an indication of the quantities the inertial sensors should be measuring, the pendulum, illustrated in figure 2, is equipped with a high-resolution, absolute encoder attached to its axis, so that the angular position of the pendulum is known and consequently, the pose of the inertial measuring unit.

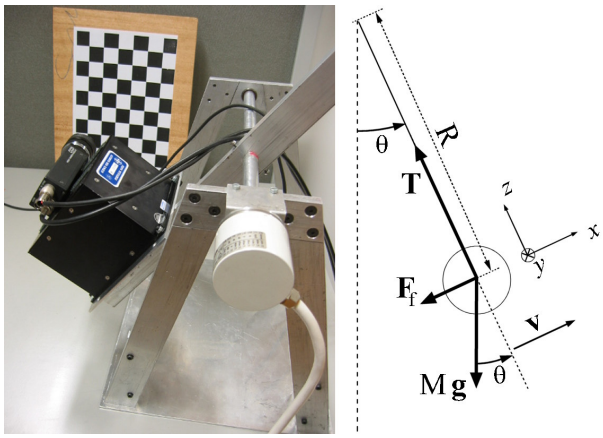


Fig. 2: *Pendulum used to calibrate the inertial sensors (on the left), schematic of the forces acting on a moving pendulum (on the right)*

On the right side of figure 2 the forces acting on the moving pendulum are represented. A friction force, $\boldsymbol{F}_f$, is represented with its direction opposite to the direction of the pendulum's instantaneous velocity, accounting for all kinds of friction inherent to the pendulum's motion.

The sum of all forces acting on the pendulum induces an acceleration which characterises the pendulum's motion equation. From this motion equation, the acceleration components along the x and z axis, as illustrated in figure 2, can be written as

$$a_x = -\|\boldsymbol{g}\| \sin\theta - \frac{\|\boldsymbol{F}_f\|}{M} sgn(\boldsymbol{v}) \quad (2)$$

$$a_z = \frac{\|\boldsymbol{T}\|}{M} - \|\boldsymbol{g}\| \cos\theta = \frac{v^2}{R} \quad (3)$$

In these equations, $sgn()$ is the signal function, given by

$$sgn(v) = \begin{cases} +1, & v \geq 0 \\ -1, & v < 0 \end{cases} \quad (4)$$

The accelerometers measure the acceleration sensed by a proof mass internal to the measuring unit which in turn is attached to the pendulum. This means that the measured accelerations are the ones caused by forces acting on the measuring unit's case, but not on the proof mass. In this particular scenario, since the gravity force acts both on the proof mass and on the case, the accelerometers only measure the accelerations caused by the other forces: the tension, $\boldsymbol{T}$, and the friction force, $\boldsymbol{F}_f$. The measured accelerations

along the x and z axis, $\tilde{a}_x$ and $\tilde{a}_z$, are given by

$$\begin{aligned} \tilde{a}_x &= -\frac{\|\boldsymbol{F}_f\|}{M} sgn(\boldsymbol{v}) = a_x + \|\boldsymbol{g}\| \sin\theta \\ &= \alpha R + \|\boldsymbol{g}\| \sin\theta \qquad (5) \\ \tilde{a}_z &= \frac{\|\boldsymbol{T}\|}{M} = \frac{v^2}{R} + \|\boldsymbol{g}\| \cos\theta \\ &= \omega^2 R + \|\boldsymbol{g}\| \cos\theta \qquad (6) \end{aligned}$$

where ω and α represent the angular velocity and angular acceleration of the pendulum.

The variables θ, ω and α are obtained by the encoder readings, and its derivatives. The measurements of the rate gyros, are the components of the angular velocity of the pendulum, meaning that the only rate gyro with a non-zero measurement should be the one oriented perpendicularly to the plane of motion. Using figure 2 as a reference, only the rate gyro along the y axis should measure a non-zero quantity, i.e.

$$\tilde{\boldsymbol{\omega}} = \begin{bmatrix} 0 \\ \tilde{\omega}_y \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ -\frac{d\theta}{dt} \\ 0 \end{bmatrix} \qquad (7)$$

By attaching the measuring unit to the pendulum in three orthogonal poses, sufficient data can be collected to calibrate the three accelerometers and the three rate gyros of the inertial measuring unit. The procedure consists in determining the nine scale factors, s_{kl}, and the three bias, b_k, of the sensor model described in (1). In order to do so, let us rewrite the system of equations (1) as a function of these unknowns. The resulting system of equations is given by

$$\begin{aligned} \boldsymbol{z}_o &= \boldsymbol{A} \cdot \boldsymbol{M'} \\ &= \begin{bmatrix} z_{i,x} & 0 & 0 \\ z_{i,y} & 0 & 0 \\ z_{i,z} & 0 & 0 \\ 0 & z_{i,x} & 0 \\ 0 & z_{i,y} & 0 \\ 0 & z_{i,z} & 0 \\ 0 & 0 & z_{i,x} \\ 0 & 0 & z_{i,y} \\ 0 & 0 & z_{i,z} \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}^T \begin{bmatrix} s_{xx} \\ s_{xy} \\ s_{xz} \\ s_{yx} \\ s_{yy} \\ s_{yz} \\ s_{zx} \\ s_{zy} \\ s_{zz} \\ b_x \\ b_y \\ b_z \end{bmatrix} \qquad (8) \end{aligned}$$

where $\boldsymbol{M'}$ is the vector with the twelve parameters to be determined.

Each measurement provides three equations as can be seen in (8). The sensor inputs, $\boldsymbol{z}_i$, are known by feeding the encoder readings, and its derivatives, into equations (5), (6) and (7); the sensor outputs, $\boldsymbol{z}_o$, are directly measured. Only the twelve parameters in vector $\boldsymbol{M'}$ are unknown. To obtain a solution for

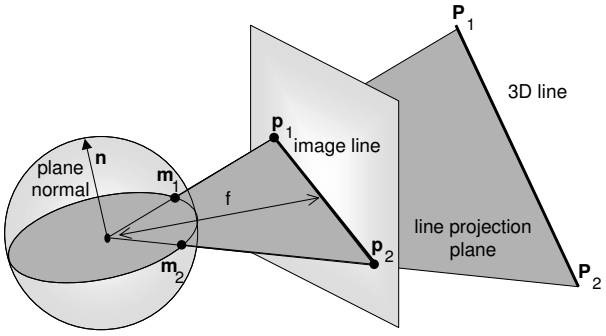


Fig. 3: *Line projection onto the Unit Sphere.*

$\boldsymbol{M'}$, at least four measurements have to be known, but since the measurements are disturbed by random noise, a much bigger set of measurements should be used.

A least squares solution can be obtained for the parameters, by using equation (9), where $\boldsymbol{A}^\dagger$ denotes the pseudoinverse of matrix $\boldsymbol{A}$ obtained through the use of the singular value decomposition (see [6]).

$$\boldsymbol{M'} = \boldsymbol{A}^\dagger \cdot \boldsymbol{z}_o \qquad (9)$$

It should be noted that two systems of equations have to be solved: one to determine the parameters of the accelerometers, and another to determine the parameters of the rate gyros.

3 Data from Camera Sensor

The pinhole camera model derives from the camera's geometry, and considers the projection of world points onto a plane, but the projection need not be onto a plane. Consider a unit sphere around the optical center, with the images being formed on its surface. The image plane can be seen as a plane tangent to a sphere of radius f, the camera's focal distance, concentric with the unit sphere, as shown in figure 3. The image plane touches the sphere at the equator, and this point defines, on the image plane, the image center. Using the unit sphere gives a more general model for central perspective and provides an intuitive visualization of projective geometry [7]. It also has numerical advantages when considering points at infinity, such as vanishing points.

A world point $\boldsymbol{P}_i$ will project on the image plane as $\boldsymbol{p}_i$ and can be represented by the unit vector $\boldsymbol{m}_i$ placed at the sphere's center, the optical center of the camera. With image centered coordinates $\boldsymbol{p}_i = (u_i, v_i)$ we have

$$\boldsymbol{P}_i \rightarrow \boldsymbol{m}_i = \frac{\boldsymbol{P}_i}{||\boldsymbol{P}_i||} = \frac{1}{\sqrt{u_i^2 + v_i^2 + f^2}} \begin{bmatrix} u_i \\ v_i \\ f \end{bmatrix} \qquad (10)$$

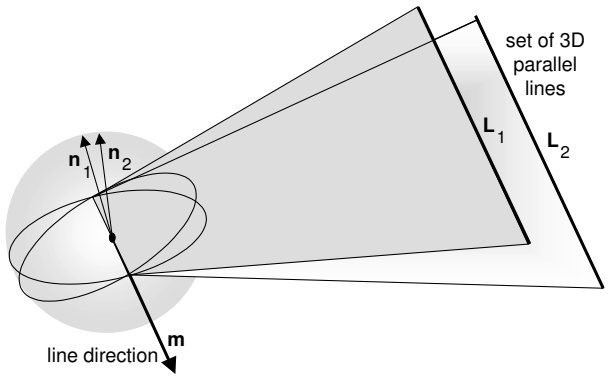


Fig. 4: *Vanishing point of a set of 3D parallel lines.*

To avoid ambiguity m_i is forced to be positive, so that only points on the image side hemisphere are considered.

Image lines can also be represented in a similar way. Any image line defines a plane with the center of projection, as shown in figure 3. A vector n normal to this plane uniquely defines the image line and can be used to represent the line.

For a given image line $au + bv + c = 0$, the unit vector is given by

$$\boldsymbol{n} = \frac{1}{\sqrt{a^2 + b^2 + (c/f)^2}} \begin{bmatrix} a \\ b \\ c/f \end{bmatrix} \tag{11}$$

As seen in figure 3, we can write the unit vector of an image line with points $\boldsymbol{m}_1$ and $\boldsymbol{m}_2$ as

$$\boldsymbol{n} = \boldsymbol{m}_1 \times \boldsymbol{m}_2 \tag{12}$$

3.1 Vanishing Points

Parallel lines only meet at infinity, but in the image plane, the point where they meet can be quite visible and is called the *vanishing point* of that set of parallel lines.

A space line with the orientation of an unit vector $\boldsymbol{m}$ has, when projected, a *vanishing point* with unit sphere vector $\pm\boldsymbol{m}$, as shown in figure 4. Since the vanishing point is only determined by the 3D orientation of the space line, projections of parallel space lines intersect at a common vanishing point.

As seen in figure 4, the normals to the line projection planes will all lie in the same plane, orthogonal to the vanishing point $\boldsymbol{m}$. The vanishing point of a set of 3D parallel lines with image lines $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$ is given by

$$\boldsymbol{m} = \boldsymbol{n}_1 \times \boldsymbol{n}_2 \tag{13}$$

4 Rotation between Camera and IMU Frames of Reference

In order to determine the rigid transformation between the INS frame of reference $\{\mathcal{I}\}$ and the camera frame of reference $\{\mathcal{C}\}$, both sensors are used to measure the vertical direction, as shown in figure 1. When the IMU sensed acceleration is equal in magnitude to gravity, the sensed direction is the vertical. For the camera, either using a specific calibration target, such as a chessboard placed vertically, or assuming the scene has enough predominant vertical edges, the vertical direction can be taken from the corresponding vanishing point.

If n observations are made for distinct camera positions, recording the vertical reference provided by the inertial sensors and the vanishing point of scene vertical features, the absolute orientation can be determined using Horn's method [8]. Since we are only observing a 3D direction in space, we can only determine the rotation between the two frames of reference.

Let ${}^{\mathcal{I}}\boldsymbol{v}_i$ be a measurement of the vertical by the inertial sensors, and ${}^{\mathcal{C}}\boldsymbol{v}_i$ the corresponding measurement made by the camera derived from some scene vanishing point. We want to determine the unit quaternion $\mathring{\boldsymbol{q}}$ that rotates inertial measurements in the inertial sensor frame of reference $\{\mathcal{I}\}$ to the camera frame of reference $\{\mathcal{C}\}$. In the following equations, when multiplying vectors with quaternions, the corresponding imaginary quaternions are implied. We want to find the unit quaternion $\mathring{\boldsymbol{q}}$ that maximizes

$$\sum_{i=1}^{n} (\mathring{\boldsymbol{q}}\, {}^{\mathcal{I}}\boldsymbol{v}_i\, \mathring{\boldsymbol{q}}^*) \cdot {}^{\mathcal{C}}\boldsymbol{v}_i \tag{14}$$

which can be rewritten as

$$\sum_{i=1}^{n} (\mathring{\boldsymbol{q}}\, {}^{\mathcal{I}}\boldsymbol{v}_i) \cdot ({}^{\mathcal{C}}\boldsymbol{v}_i\, \mathring{\boldsymbol{q}}) \tag{15}$$

The quaternion product can be expressed as a matrix. With ${}^{\mathcal{I}}\boldsymbol{v}_i = ({}^{\mathcal{I}}x_i, {}^{\mathcal{I}}y_i, {}^{\mathcal{I}}z_i)^T$ and ${}^{\mathcal{C}}\boldsymbol{v}_i = ({}^{\mathcal{C}}x_i, {}^{\mathcal{C}}y_i, {}^{\mathcal{C}}z_i)^T$ we have

$$\mathring{\boldsymbol{q}}\, {}^{\mathcal{I}}\boldsymbol{v}_i = \begin{bmatrix} 0 & -{}^{\mathcal{I}}x_i & -{}^{\mathcal{I}}y_i & -{}^{\mathcal{I}}z_i \\ {}^{\mathcal{I}}x_i & 0 & {}^{\mathcal{I}}z_i & -{}^{\mathcal{I}}y_i \\ {}^{\mathcal{I}}y_i & -{}^{\mathcal{I}}z_i & 0 & {}^{\mathcal{I}}x_i \\ {}^{\mathcal{I}}z_i & {}^{\mathcal{I}}y_i & -{}^{\mathcal{I}}x_i & 0 \end{bmatrix} \mathring{\boldsymbol{q}} = {}^{\mathcal{I}}\mathbb{V}_i \mathring{\boldsymbol{q}} \tag{16}$$

and

$${}^{\mathcal{C}}\boldsymbol{v}_i\, \mathring{\boldsymbol{q}} = \begin{bmatrix} 0 & -{}^{\mathcal{C}}x_i & -{}^{\mathcal{C}}y_i & -{}^{\mathcal{C}}z_i \\ {}^{\mathcal{C}}x_i & 0 & -{}^{\mathcal{C}}z_i & {}^{\mathcal{C}}y_i \\ {}^{\mathcal{C}}y_i & {}^{\mathcal{C}}z_i & 0 & -{}^{\mathcal{C}}x_i \\ {}^{\mathcal{C}}z_i & -{}^{\mathcal{C}}y_i & {}^{\mathcal{C}}x_i & 0 \end{bmatrix} \mathring{\boldsymbol{q}} = {}^{\mathcal{C}}\mathbb{V}_i \mathring{\boldsymbol{q}} \tag{17}$$

$$\boldsymbol{N} = \begin{bmatrix} (S_{xx}+S_{yy}+S_{zz}) & S_{yz}-S_{zy} & S_{zx}-S_{xz} & S_{xy}-S_{yx} \\ S_{yz}-S_{zy} & (S_{xx}-S_{yy}-S_{zz}) & S_{xy}+S_{yx} & S_{zx}+S_{xz} \\ S_{zx}-S_{xz} & S_{xy}+S_{yx} & (-S_{xx}+S_{yy}-S_{zz}) & S_{yz}+S_{zy} \\ S_{xy}-S_{yx} & S_{zx}+S_{xz} & S_{yz}+S_{zy} & (-S_{xx}-S_{yy}+S_{zz}) \end{bmatrix} \quad (23)$$

Substituting in (15)

$$\sum_{i=1}^{n} ({}^{\mathcal{I}}\mathbb{V}_i \, \mathring{\boldsymbol{q}}) \cdot ({}^{\mathcal{C}}\mathbb{V}_i \, \mathring{\boldsymbol{q}}) \quad (18)$$

or

$$\sum_{i=1}^{n} \mathring{\boldsymbol{q}}^T \, {}^{\mathcal{I}}\mathbb{V}_i{}^T \, {}^{\mathcal{C}}\mathbb{V}_i \, \mathring{\boldsymbol{q}} \quad (19)$$

factoring out $\mathring{\boldsymbol{q}}$ we get

$$\mathring{\boldsymbol{q}}^T \left(\sum_{i=1}^{n} {}^{\mathcal{I}}\mathbb{V}_i{}^T \, {}^{\mathcal{C}}\mathbb{V}_i \right) \mathring{\boldsymbol{q}} \quad (20)$$

So we want to find $\mathring{\boldsymbol{q}}$ such that

$$\max \mathring{\boldsymbol{q}}^T \, \boldsymbol{N} \, \mathring{\boldsymbol{q}} \quad (21)$$

where $\boldsymbol{N} = \sum_{i=1}^{n} {}^{\mathcal{I}}\mathbb{V}_i{}^T \, {}^{\mathcal{C}}\mathbb{V}_i$. Having

$$S_{xx} = \sum_{i=1}^{n} {}^{\mathcal{I}}x_i \, {}^{\mathcal{C}}x_i \, , \; S_{xy} = \sum_{i=1}^{n} {}^{\mathcal{I}}x_i \, {}^{\mathcal{C}}y_i \quad (22)$$

and analogously for all 9 pairings of the components of the two vectors, matrix $\boldsymbol{N}$ can be expressed using these sums as in (23). The sums contain all the information that is required to find the solution.

Since $\boldsymbol{N}$ is a symmetric matrix, the solution to this problem is the four-vector $\boldsymbol{q}_{max}$ corresponding to the largest eigenvalue λ_{max} of $\boldsymbol{N}$ - see [8] for details.

5 Results

5.1 Inertial Sensor Calibration

The inertial sensors were calibrated using several measurements from both the inertial sensors and the absolute encoder. The inertial measurement unit was attached to the pendulum in three distinct orientations in order to measure significative quantities for each sensor. The sensors' outputs were logged, and the inputs were estimated by using the encoder readings to feed equations (5), (6) and (7).

Since the inertial measurement unit used in this work is a medium-grade unit, the manufacturer supplies an individual calibration table which can be used as a ground truth to evaluate our calibration procedure.

Table 1 presents the parameters supplied by the manufacturer and compares them to the ones obtained using the calibration method described in this paper. It should be noted that in the table, the sensitivity is compared in (g/V), which is the inverse of the scale factors, s_{kk}, as defined in (1).

Table 1: *Comparison of the obtained inertial sensors' parameters with the ones supplied by the manufacturer.*

Accelerometers				
	Sensitivity (g/V)		Null Offset (V)	
Axis	Supplied	Obtained	Supplied	Obtained
X	1.008	0.9987	2.485	2.5259
Y	1.000	1.0273	2.519	2.5095
Z	1.017	1.0295	2.455	2.4459
Rate Gyros				
	Sensitivity (g/V)		Null Offset (V)	
Axis	Supplied	Obtained	Supplied	Obtained
X	102.731	102.1154	2.499	2.5015
Y	101.643	102.1553	2.499	2.4997
Z	102.388	102.0536	2.499	2.5001

The manufacturer doesn't present any parameters relating to axis alignment in their unit, but our results show a mean cross-axis sensitivity of about 0.6%.

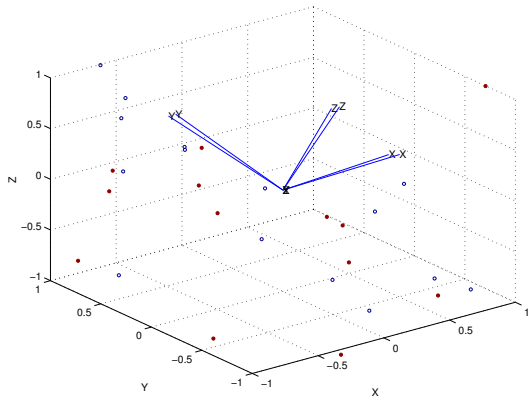
The results obtained by this process are satisfactory since they only differ slightly from the ones supplied by the manufacturer. On the other hand, even these small differences can cause high drifts over time if the measurements are to be used to estimate position by integrating the sensors data.

5.2 Rotation Estimation

To validate the above method, a simulation was performed, using a known rotation with added noise. Figure 5 shows the data set with added noise used, the known rotation, and the estimated rotation. The adjoining table indicates the estimated quaternions and the angular errors. With improving signal to noise ratio, the estimated rotation approaches the real value.

The camera calibration toolbox provided by Intel Open Source Computer Vision Library [9] was used to provide a standard camera calibration method. The calibration used images of a chessboard target in several positions and recovers the camera's intrinsic parameters, as well as the target positions relative to the camera. The calibration algorithm is based on Zhang's work in estimation of planar homographies for camera calibration [10].

In a test sequence, the camera was moved with the target in sight, and all IMU data and images logged. The camera calibration was performed with images sampled from the complete set recorded. Figure 6



20 random positions, rotated with added noise		
real	$0.98079 < 0.18110, 0.040244, 0.060366 >$	error(deg)
SNR=5	$0.97568 < 0.18435, 0.065185, 0.099079 >$	5.31°
SNR=10	$0.97702 < 0.19855, 0.062037, 0.046521 >$	3.57°
SNR=20	$0.98476 < 0.15888, 0.041402, 0.057394 >$	2.58°
SNR=40	$0.98072 < 0.18157, 0.040012, 0.060126 >$	0.07°

Fig. 5: *Plot of simulation data for 20 random positions, with added noise SNR=5, showing the known and estimated rotation, and table with simulation results, indicating angular error of the estimated rotations.*

shows some of the images used and the reconstructed camera positions.

Having calibrated the camera, the chessboard target was placed vertical and the vertical vanishing point determined, providing a set of measurements ${}^{\mathcal{C}}\boldsymbol{v}_i$. Having the corresponding ${}^{\mathcal{I}}\boldsymbol{v}_i$ from the inertial sensors, the estimation method was applied to the data set. Figure 6 shows the result obtained for a real data set. The estimated rotation has an angle 91.25° about an axis $(0.89, -0.27, -0.3582)$, and is the near expected one, given the mechanical mount, of a near right angle approximately about the x axis. Re-projecting the inertial sensor data showed consistency of the method. The mean-square error in the re-projected unit vectors was 0.2019° with variance $\sigma^2 = 1.5699°$.

6 Conclusions

This paper presented a method for inertial sensor integration in vision systems. In order to use off-the-shelf inertial sensors and cameras, appropriate modelling and calibration techniques were presented.

Using a pendulum with an encoded shaft, inertial sensor alignment, bias and scale factor can be estimated, for both accelerometers and gyros. With the inertial sensors rigidly fixed to the camera, the rotation between the two frames of reference can be found by moving the system and observing the vertical direction with both sensors. The inertial sensors when static only sense gravity, providing a vertical reference. Sets of parallel vertical edges provide the vertical vanishing point, giving the vertical direction in the camera frame of reference. The two sets of measurements allow the estimation of the rotation between the sensors. Knowing this rotation, the inertial sensor data can be mapped to the camera frame of reference, and used in image processing tasks [2][3][4][5][11].

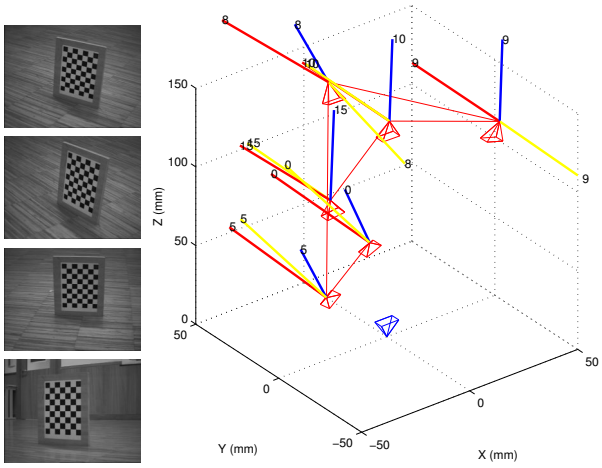


Fig. 6: *Some of the images used and result obtained, showing camera position, verticals and re-projected vectors.*

References

[1] *CyberCars – Cybernetic Technologies for the Car in the City*, http://www.cybercars.org.

[2] Jorge Lobo and Jorge Dias, "Integration of Inertial Information with Vision towards Robot Autonomy," in *Proceedings of the 1997 IEEE International Symposium on Industrial Electronics*, Guimaraes, Portugal, July 1997, pp. 825–830.

[3] Jorge Lobo and Jorge Dias, "Ground plane detection using visual and inertial data fusion," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems - IROS'98*, Victoria, Canada, October 1998, pp. 912–917.

[4] Jorge Lobo and Jorge Dias, "Fusing of image and inertial sensing for camera calibration," in *Proceedings of the International Conference on Multisensor Fusion and Integration for Intelligent Systems, MFI 2001*, Baden-Baden, Germany, August 2001, pp. 103–108.

[5] Jorge Lobo, Carlos Queiroz, and Jorge Dias, "Vertical world feature detection and mapping using stereo vision and accelerometers ," in *Proceedings of the 9th International Symposium on Intelligent Robotic Systems - SIRS'01*, Toulouse, France, July 2001, pp. 229–238.

[6] Richard Hartley and Andrew Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2000, ISBN 0521623049.

[7] Kenichi Kanatani, *Geometric Computation for Machine Vision*, Oxford University Press, 1993, ISBN 0-19-856385-X.

[8] Horn, B.K.P., "Closed-Form Solution of Absolute Orientation Using Unit Quaternions", *Journal of the Optical Society of America*, Vol. 4, No. 4, April 1987, pp.629–642.

[9] Intel, "Intel Open Source Computer Vision Library," http://www.intel.com/ research/ mrl/research/opencv/, 2003.

[10] Z. Zhang, "Flexible Camera Calibration By Viewing a Plane From Unknown Orientations," in *Proceedings of the Seventh International Conference on Computer Vision (ICCV'99)*, Kerkyra, Greece, September 1999, pp. 666–673.

[11] Jorge Lobo, "Inertial Sensor Data Integration in Computer Vision Systems," M.S. thesis, University of Coimbra, April 2002.