IST 2009 - International Workshop on Imaging Systems and Techniques
Shenzhen, China
May 11-12, 2009

Pose Estimation of a Volant Platform With a Monocular Visuo-Inertial System

Nikolaos Kyriakoulis[1], Evangelos Karakasis[1], Antonios Gasteratos[1], and Angelos Amanatiadis[2]

[1]Laboratory of Robotics and Automation, Department of Production and Management Engineering,
Democritus University of Thrace, University Campus Kimmeria, GR-67100, Xanthi, Greece
E–mail: {nkyriako, ekarakas, agaster}@pme.duth.gr
[2]Laboratory of Electronics, Department of Electrical and Computer Engineering,
Democritus University of Thrace, 12 Vas. Sofias Str., GR-67100, Xanthi, Greece
Email: aamanat@ee.duth.gr

***Abstract*—One of the serious problems in robotics applications is the estimation of the robot's pose. A lot of research effort has been put on finding the pose via inertial and proximity sensors. However, the last decades many systems adopt vision to estimate the pose, by using homographies and projection geometry. In this paper the pose estimation is achieved by the identification of a geometrically known platform from one camera and from the measurements of an inertial unit. The extended Kalman filter (EKF) is used for data fusion and error compensation. The novelty of this system is that the visual sensor and the inertial unit are mounted on different mobile systems. The proposed pose estimation system exhibits high accuracy in real-time.**

***Index Terms*—Visual pose estimation, inertial pose estimation, visuo-inertial fusion.**

I. Introduction

The estimation of the pose through visual data is an active research field and much effort has been applied for its cooperation with the inertial unit sensors. In most biomimetic robotic systems, the cameras serve the role of the eyes while the inertial sensor the vestibule. The use of inertial unit sensors for the pose estimation problem has many advantages as the camera measurements are complementary with the inertial ones. The high frequency gyro measurements can be combined with the low frequency visual ones, providing high frequency pose estimates [1]. The integration of visual and inertial measurements is very attractive due to its complementarity, robustness as well as its flexibility. The pose estimation problem using visual sensors requires image processing, that is known to be computationally expensive. The need of fast and accurate vision systems has led to the adoption of cameras with high frame rate. Still, the frequency of the inertial units is much higher than the one of the visual systems. Nevertheless, the inertial units introduce accumulative error into their measurements. The challenge is the integration of these sensors, while achieving balance between accuracy and processing time.

In this paper our goal is to estimate the pose of an indoors volant platform. A visual sensor is mounted on a moving unit on the ceiling and it is fixated towards the floor. The volant platform is hanged from the ceiling unit. On its top surface there are patterns with known geometry, which are identified by the visual sensor. An inertial unit sensor is mounted on the ceiling unit, and another one on the volant platform. For comprehensive reasons the two sensors are going to be referred as $inertial_c$ and $inertial_v$, for the inertial unit on the ceiling unit and the volant platform, respectively. The data from the camera and the two inertial units are fused in two concurrent steps. The first one includes the integration of the $inertial_c$ with the camera in order to compute the camera's accurate position. The second step involves the fusion of the ceiling unit pose estimation with the $inertial_v$ measurements. We had to overcome many difficulties as both sensors are moving while having different frames of reference. For the data fusion we used the extended Kalman filter (EKF) due to its recursive nature, and its capability to be applied to non-linear problems [2]. First the synchronization of the sensors occurs and then the fusion of their data. In order to meet the requirements of low processing times the proposed system utilizes the C# OpenCV (Open Source Computer Vision) libraries [3] for the vision tasks, i.e. camera calibration, feature extraction, and pose estimation, which exhibit low computation demands and, thus, they achieve real-time operation. Regarding the inertial units' readings, a C# algorithm was implemented, as well. The final fusion between the measurements of the ceiling unit with the volant platform's one is the output of our system. The provided results are very accurate and the whole processing is realized in real-time. The proposed system is going to be implemented within the framework of the ACROBOTER European research project[1].

A. Related Work

The pose estimation problem with visual sensors concerns geometrical references between the 3D object and the respective projections onto the 2D image plane [4]. Therefore, the pose estimation problem has been approached by geometrical projections [5], [6]. Moreover, Gallagher *et al.* [7] made use of the knowledge that the parallel lines of the real world are

[1]The ACROBOTER project is still under development. There are copyright restrictions for demonstrating the proposed parts into images. The only parts that can be illustrated are the camera and the inertial sensors.

978-1-4244-3483-1/09/$25.00 ©2009 IEEE

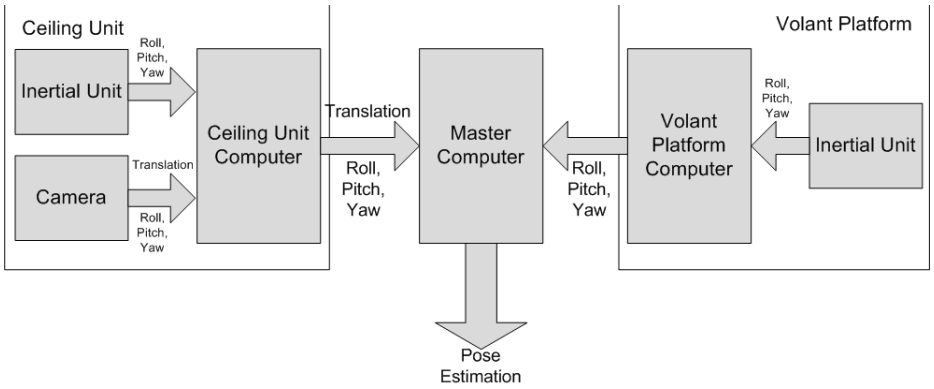


Fig. 1. The proposed system architecture.

connecting to a point in the image plane, which is called vanishing point. In an image there can be more than one vanishing points, the identification of which leads to the extraction of the rotation matrix [7] and the absolute orientation of an object [8]. In [9] the horizon line is determined by inertial sensors, and by using one vanishing point and the vertical reference, the camera's pose is estimated.

Besides the geometrical approaches, the pose estimation problem has been transformed into a minimization problem of the object-space function [4]. Furthermore, a solution to the pose estimation problem has been given, using a planar target, by geometric interpretation and by finding two local minima of the object-space error function [10]. The pose estimation can also be achieved by a set of algorithms form n points or n lines by linearizing quadratic systems [11]. However, most of the pose estimation approaches present high degree of error at the final transformation matrix. The reduction of these errors optimizes the accuracy of the final pose estimation [12].

There are three main visuo-inertial integration categories, the correction, the colligation and the fusion one [13]. The correction integration type deals with considering the measurements of one sensor as the ground truth, whereas the other measurements as the estimated ones. The goal is to reduce the error between both measurements to the minimum. In the case of colligation the measurements of both sensors are combined. That is, if the visual data is used for estimating the pose, the inertial one will be used for the control to verify these measurements [14]. Finally, the fusion of the visual and inertial sensors takes into account both measurements. Gemeiner *et al.* have estimate the egomotion and the environmental structure by a binocular vision system integrated with an inertial sensor [15]. Shademan *et al.* [16] have made a comparative study for fusing inertial sensors with visual ones via EKF and iterated extended Kalman filter (IEKF). The chosen fusion method plays a key role to the final estimation as it is highly application dependent.

The presented system is similar to the aforementioned work. The main difference however is that the visual sensor is mounted outside the system and, also, its reference frame is a different one. Furthermore, as the visual sensor moves on the ceiling, while the robot performs a free motion in the 3D space, i.e. in a 6 d.o.f. space, the complexity of the system is increased significantly. The architecture of the presented system is novel with respect to the fact that the visual sensor lies outside the robot's system.

II. System Architecture

The proposed system includes two computers for the respective different systems, i.e. the ceiling pose estimation unit and the volant one, both of which communicate with a master computer. The sensorial system of the ceiling unit comprises a visual sensor and an inertial unit. Based on the pinhole camera model, the visual sensor computes the transformation matrix by multiplying the intrinsic parameters matrix with the image feature's coordinates. The computed rotation and translation matrices from the visual sensor, along with the inertial units ones, are sent to the respective computers, and finally to the master computer for further processing. The fusion of the pose estimation is performed at the master computer. The two computers are connected with the master one, through a high speed network protocol. The system architecture is capable to estimate the pose accurately, in real-time. Figure 1 illustrates the system's architecture in a block diagram.

A. Hardware Setup

The sensory configuration of the proposed system consists of one camera and two inertial unit sensors. The camera is the Pointgrey Grasshopper, whilst both inertial units are the MTx Xsens. The interfaces used are IEEE 1394 for the camera and USB 2.0 to serial outputs for the inertial unit sensors. The visual sensor and one of the inertial units are mounted on a ceiling unit. Furthermore, the camera is placed so as its optical axis to be perpendicular to the ground, as shown in Fig. 2.

The communication protocol between the computers is built on top of sockets, meeting the requirements for low latency. The synchronization of the two computers with the master one is achieved by thread programming. There are two processes that are running simultaneously. The first process is the transformation matrix computation from the inertial unit on the ceiling system and the visual sensor. The rotations

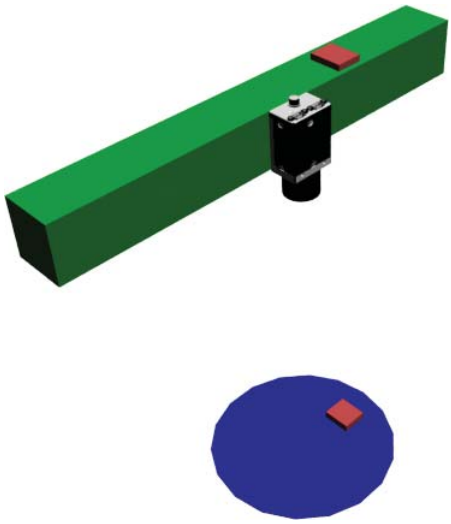


Fig. 2. The proposed system hardware architecture. The orange boxes represent the inertial units for the ceiling unit, green parallelepiped, and the volant platform, blue cylinder, respectively.

at the ceiling unit are mostly around the Roll axis, as the movements are restricted to be parallel to the ground. The integration of both measurements is realized by multiplying the inertial units measurements to the camera's rotation matrix. The synchronization of these measurements is succeeded by simply decreasing the frequency of the inertial unit down to the camera's one. The translation matrix is calculated only by the visual measurements. Finally, the acquired transformation matrix is sent to the master computer. The second process is the extraction of the volant platform's rotation matrix from the respective inertial unit. Subsequently, the read angles are sent to the master computer, where the results from both processes are fused.

The visual pose estimation process demands high accuracy at the image feature recognition. For the determination of the internal camera geometric and optical characteristics, we utilized the camera calibration method available in the OpenCV library. Calibration is also needed for the inertial unit sensors, due to the electromagnetic noise introduced by the cables and the camera.

B. Software Setup

The sensory drivers are used to read the respective measurements. They provide all the necessary settings for optimal operation such as the frame grabbing, the resolution changes and the frame rate concerning the camera, or the orientation output type, and the frequency regarding the inertial unit sensors.

In order achieve a higher frame-rate (in particular 30fps), the chosen resolution is 640×480. In each captured image the features are identified and the extrinsic parameters are computed. The OpenCV libraries were used for calculating the homography matrices. There are five different frames of reference, the image $\{Im\}$; the camera $\{Cm\}$; the *inertial*$_c$ $\{Cin\}$; the *inertial*$_v$ $\{Vin\}$; and the volant platform's $\{Vp\}$ one. The two inertial unit sensors have the same orientation with the volant platform, while the $\{CM\}$ one is rotated by -180° around the *x* axis at $\{Cin\}$. In order to apply the same orientation at $\{Cm\}$ and $\{Im\}$, the z axis at $\{Cm\}$ is further rotated by -90°. The final camera's position is given by Eq.(1) and (2)

$$R_{Cm} = R_{(-180,0,0)} R_{(0,0,-90)} R_{Cin} \tag{1}$$

$$R_{Cm} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} R_{Cin} \tag{2}$$

The rotation angles calculated from the matrix in Eq. (2) are directly sent to the master computer.

Considering the volant platform subsystem, the procedure is much simpler, as it possesses the same orientation with the inertial unit mounted on it and is the only present sensory. The frequency of this unit is higher than the ceiling unit's one, as there is no vision input to be fused with. The measurements are preserved into a buffer, and ultimately sent to the master computer when the process of the ceiling unit is concluded. The frequency is intentionally kept high for redundancy reasons, as there might be occlusions, meaning failure to the vision system. The measurements of both units are logged files for later testing.

An important role for the real-time operation of the proposed system is the communication protocol and the adoption of C# as programming language. The master computer receives data from both units simultaneously by thread programming. Two threads are developed in the respective computers and a master one is designed to control the other two. The communication is ended when both measurements are available at the master computer.

The received measurements are fused with the utilization of an extended Kalman filter (EFK). The mathematical formulation of the EFK is described in [2]. The final system's output regarding the rotation angles, has to follow the *inertial*$_v$ measurement, in case of occlusions presence to the vision system. The tuning variables *Q* and *R* for the process and the measurement noise, respectively, are set to a ratio of 10 (*R*/*Q*= 10). The output is closer to the inertial measurements by decreasing the value of *Q*. On the contrary, when *R* is less then the output is closer to the camera measurements. The measurement noise covariance, *R*, defines the level that the filter follows the measurements. In case of wishing the output to follow the camera estimates in higher degree, the value of *R* should be further decreased. The process noise covariance, *Q*, represents the error introduced during the process, thus, in case of having reliable measurement values, it should be increased. The higher the ratio *R*/*Q* is, the more unreliable are considered the camera measurements and the output is closer to the *inertial*$_v$ measurements. The fine tuning of the variables is a trial and error process.

Fig. 3. The camera mounted onto a pan-tilt mechanism for the assessment of the vision system.

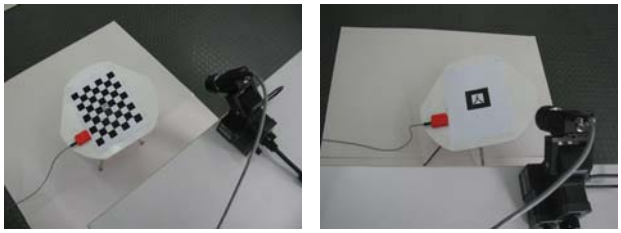
TABLE I
RESPONSE TIMES FOR ALL SUBSYSTEMS.

Subsystem	Time (ms)
Vision System	6.65
Ceiling's Inertial	0.03
Platform's Inertial	0.01
Latency	0.02
Fusion	0.01
Overall	6.72

III. EXPERIMENTAL RESULTS

The goal of the proposed system is to measure the pose of the volant platform in real-time. Moreover, the pose estimation has to exhibit high accuracy in real-time, which is achieved by employing the OpenCV libraries. The average response time for the vision system, from its initialization until the initial pose estimation is 6.65msec. The time needed for capturing an image from the camera is 1.01ms, while the feature extraction part lasts 5.20ms. The final step is to calculate the platform's pose from the extracted features. This procedure lasts only 0.43ms, as it involves simple mathematical calculations. The inertial units correspondences are 30Hz for the ceiling unit and 100Hz for the volant platform. Table I summarizes the average response times of each subsystem along with the overall one. The values are counted in msecs and correspond to the time needed for a single pose estimate.

Despite the low response times, the vision system exhibits high accuracy. In order to evaluate the rotation measurements, the camera was mounted onto the pan-tilt mechanism as shown in Fig. 3. We set known angles to the mechanism and rotated the volant platform randomly, while storing all the measurements. The distances from the camera were also arbitrarily selected in order to examine the robustness of the translation estimation. The performance of the vision system was further testified by assessing the ARToolKit [17] instead of the OpenCV, to obtain the camera's pose. ARToolkit is usually used as benchmark for the estimated pose from planar targets [10]. We run a similar test, due to the random rotations and translations, with the pan-tilt mechanism. The pose estimates were logged and compared with the ones from the OpenCV.



(a) (b)

Fig. 4. The camera mounted onto the pan-tilt mechanism recognizing (a) a chessboard for the OpenCV measurements, (b) a pattern for the ARToolKit ones.

TABLE II
OVERALL RESULTS FOR ALL THE TESTED SENSORY USED. THE PAN-TILT IS USED AS THE GROUND TRUTH.

	Pan-Tilt	OpenCV	ARToolkit	In. Unit
Roll	14.66	13.99	15.27	14.83
Pitch	4.47	4.58	4.12	4.36
Yaw	9.08	8.84	9.22	8.93
Accuracy	100%	96.4%	96%	98.6%
Error	0	0.34	0.36	0.15

Figures 4(a), and 4(b) depict the system setup using the OpenCV and the ARToolKit, respectively. The differences between the mechanism's angles and the ones computed with the vision algorithms, have the same error values. As far as the translation vector computation is concerned, we used a laser sensor in order to use it as the ground truth benchmark. The error for both sets of algorithms is less than 1 degree regarding the rotations and less than 1mm concerning the translations.

Table II contains the performance of the vision tools, i.e. OpenCV and ARToolikit, in relationship with the volant platform's inertial unit and the pan-tilt mechanism. The reason that the pan-tilt has zero error and 100% accuracy, is because its output is considered to be the ground truth to all measurements, as it has the highest accuracy and can be easily manipulated to be used as benchmark. Table III presents the translation measurement values,which are measured in mm. The laser values have 100% accuracy and zero error because it is considered as the ground truth benchmark. The results shown in Tables II and III are the mean values of about 10000 acquired measurements.

Figure 5 shows a plot diagram of the experimental results. The green line represents the master computer's final output, which is the fused pose estimate. The blue line corresponds to the volant platform measurements, whilst the red one to the

TABLE III
OVERALL RESULTS FOR ALL THE TESTED SENSORY USED. THE LASER IS USED AS THE GROUND TRUTH.

Axes	Laser	OpenCV	ARToolkit
x	56.21	55.87	56.97
y	34.82	33.31	35.24
z	76.63	75.97	76.08
Accuracy	100%	98.06%	98.91%
Error	0	0.83	0.57

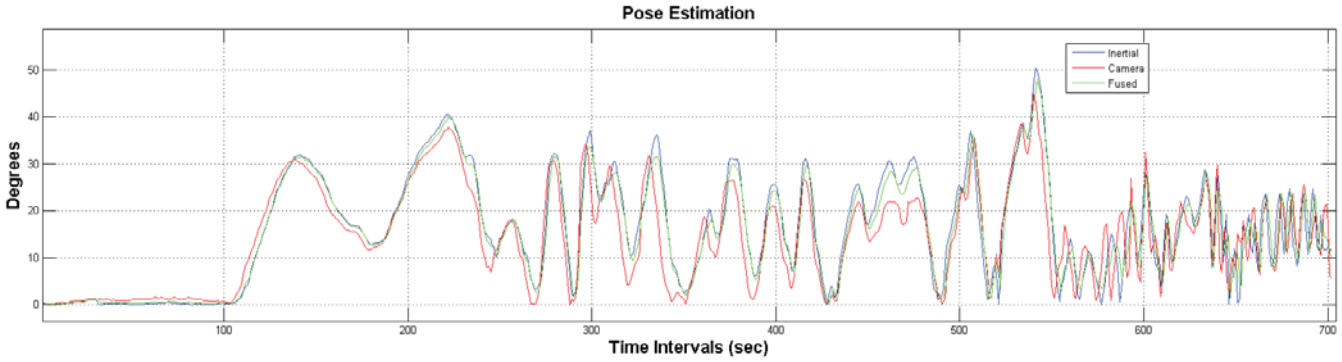


Fig. 5. The response of each subsystem. The blue line represents the pose estimation of the volant platform. The red line corresponds to the final camera pose estimation.

estimated pose from the ceiling unit. The ARToolkit output is excluded from the diagram as it is not used to the system. It is obvious that the EFK application improves the efficiency of our system. The final output is closer to the inertial unit measurements than the camera, due to the tuning variables ratio. Although the vision system exhibits high performance in long-lasting tasks, the inertial unit is more robust, as it provides more measurements, without discontinuities caused by obstacles intruding to the image field. Thus, for safety and redundancy reasons the inertial unit gains more weight to its measurements during the fusion process.

IV. Discussion and Conclusions

In this paper a pose estimation system has been demonstrated aimed for autonomous robotics applications. The system utilizes two inertial unit sensors and one camera. There are two subsystems, the ceiling unit and the volant platform one. The ceiling unit consists of the camera and one inertial unit, while the volant platform of one inertial unit. The aim is to fuse both measurements in order to obtain the platform's pose in real-time. On each step the ceiling unit's inertial sensors provide the camera's accurate position. The ceiling unit's measurements are integrated at the respective computer, and are sent to the master computer for fusion with the inertial unit's measurements from the platform subsystem. A fusion process by assessing an extended Kalman filter provides the final accurate pose estimation.

The requirement of low processing times is met by using the OpenCV libraries regarding the vision system, which is the most demanding in terms of computational resources. The thread programming assures the simultaneous operation of the two subsystems. The inertial unit sensors operate in high frequencies without applying more computational load to the system. Finally, after the utilization of the extended Kalman filter, the final output of the system is available every 6.72msec as shown in Table II.

This work proves the possibility to obtain a real-time pose estimate of two systems, which have different frames of reference. The identification of a robot's pose, with a camera which is not mounted onto it, provides flexibility to the system. Thus, it can be used to estimate any robot platform or arm, even one performing an operation. The goal to the most autonomous robotic applications is to estimate the robot's pose in real-time. Sometimes, there are further restrictions to the size and the weight payload. In these cases the proposed pose estimation architecture is appealing, as the only sensory mounted is an inertial unit. Furthermore, the control of such a complex problem, demands high accuracy and redundancy provided from more than one sensors, making the proposed architecture ideal. However, the cost of the presented architecture is quite high, as there are three computers, and three sensor units. Though, depending on the application the cost can be reduced by employing only one inertial unit, with a steady camera. Moreover, the cost can be further be restricted by applying all sensory to one computer, resulting so in a low cost efficient pose estimation system. Nowadays, the structure of the demonstrated architecture can be easily implemented in a single computer, instead of three, as there are available quad-core processors. Each distinct procedure can be assigned to a core, while the forth will supervise the whole progress.

The system's architecture bears high optimization potential. The operating systems can be changed to a Real-Time one, which will provide lower processing times. The accuracy can be improved by adopting a more sophisticated control during the fusion process.

Acknowledgment

This work is supported by the E.C. under the FP6 research project for Autonomous Collaborative Robots to Swing and Work in Everyday EnviRonment ACROBOTER, FP6-IST-2006-045530.

References

[1] H. Rehbinder and B. Ghosh, "Pose estimation using line-based dynamic vision and inertial sensors," *Automatic Control, IEEE Transactions on*, vol. 48, no. 2, pp. 186–199, Feb. 2003.

[2] G. Welch and G. Bishop, "An introduction to the kalman filter," Chapel Hill, NC, USA, Tech. Rep., 1995. Revised, July 2006.

[3] Open Source Computer Vision (OpenCV) home page, *http://sourceforge.net/projects/opencvlibrary*.

[4] C.-P. Lu, G. D. Hager, and E. Mjolsness, "Fast and globally convergent pose estimation from video images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 6, pp. 610–622, 2000.

[5] J. Y. Aloimonos, "Perspective approximations," *Image Vision Comput.*, vol. 8, no. 3, pp. 179–192, 1990.

[6] R. Horaud, F. Dornaika, and B. Lamiroy, "Object pose: The link between weak perspective,paraperspective, and full perspective," *Int. J. Comput. Vision*, vol. 22, no. 2, pp. 173–189, 1997.

[7] A. C. Gallagher, "Using vanishing points to correct camera rotation in images," in *CRV '05: Proceedings of the 2nd Canadian conference on Computer and Robot Vision*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 460–467.

[8] S. Segvic and S. Ribaric, "Determining the absolute orientation in a corridor using projective geometry and active vision," *Industrial Electronics, IEEE Transactions on*, vol. 48, no. 3, pp. 696–710, Jun 2001.

[9] J. Lobo and J. Dias, "Vision and inertial sensor cooperation using gravity as a vertical reference," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1597–1608, 2003.

[10] G. Schweighofer, "Robust pose estimation from a planar target," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2024–2030, 2006, member-Pinz,, Axel.

[11] K. C. Manmohan, S. Christoph, and P. Axel, "Real-time camera pose in a room," vol. 2626/2003, pp. 98–110, 2003.

[12] Z. Zhang, D. Zhu, and J. Zhang, "An improved pose estimation algorithm for real-time vision applications," *Communications, Circuits and Systems Proceedings, 2006 International Conference on*, vol. 1, pp. 402–406, June 2006.

[13] D. Xu and Y. F. Li, "A new pose estimation method based on inertial and visual sensors for autonomous robots," *Robotics and Biomimetics, 2007. ROBIO 2007. IEEE International Conference on*, pp. 405–410, Dec. 2007.

[14] J.-C. Zufferey and D. Floreano, "Fly-inspired visual steering of an ultralight indoor aircraft," *Robotics, IEEE Transactions on*, vol. 22, no. 1, pp. 137–146, Feb. 2006.

[15] P. Gemeiner and M. Vincze, "Motion and structure estimation from vision and inertial sensor data with high speed cmos camera," *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pp. 1853–1858, April 2005.

[16] A. Shademan and F. Janabi-Sharifi, "Sensitivity analysis of ekf and iterated ekf pose estimation for position-based visual servoing," *Control Applications, 2005. CCA 2005. Proceedings of 2005 IEEE Conference on*, pp. 755–760, Aug. 2005.

[17] ARToolKit Plus, "http://studierstube.icg.tu-graz.ac.at," 2006.