# Real-time Localization in Outdoor Environments using Stereo Vision and Inexpensive GPS

Motilal Agrawal and Kurt Konolige
SRI International
333 Ravenswood Ave
Menlo Park, CA 94025
{agrawal, konolige}@ai.sri.com

## Abstract

*We describe a real-time, low-cost system to localize a mobile robot in outdoor environments. Our system relies on stereo vision to robustly estimate frame-to-frame motion in real time (also known as visual odometry). The motion estimation problem is formulated efficiently in the disparity space and results in accurate and robust estimates of the motion even for a small-baseline configuration. Our system uses inertial measurements to fill in motion estimates when visual odometry fails. This incremental motion is then fused with a low-cost GPS sensor using a Kalman Filter to prevent long-term drifts. Experimental results are presented for outdoor localization in moderately sized environments ($\geq$ 100 meters)*

## 1 Introduction

The ability of a mobile robot to localize itself is critical to its autonomous operation and navigation. A robot that navigates using maps must be able to accurately localize itself. Consequently, there has been considerable effort on the problem of mobile robot localization and mapping. This problem is known as simultaneous localization and mapping (SLAM) and there is a vast amount of literature on this topic (see e.g., [21] for a comprehensive survey). SLAM has been especially succesful in indoor structured environments [9, 7, 20].

On the other hand, localization and mapping for outdoor environments is still an open research problem. For outdoor environments, accurate localization can be obtained by using differential GPS and/or high-quality, expensive inertial navigation systems. However, high cost and size of these systems limit their applicability to smaller robotic platforms. SLAM can be performed using different types of sensors such as sonar [20], laser range finders [22] and vision [18, 4, 19]. Sonar is fast and cheap but usually very crude. Laser range scanners are accurate but slow and bulky. Vision systems are light-weight, compact, relatively inexpensive and can provide high resolution images for localization as well as mapping at a fairly high frequency.

Our goal for this project was to develop an inexpensive localization system using stereo vision and complement it with a low-cost GPS sensor and a suite of inexpensive inertial navigation sensors. The stereo vision system is designed so as to provide obstacle detection capabilities over a few meters in front of the camera in addition to providing localization capability.

Figure 1 shows our robot setup. The main perception sensors are a pair of Bumblebee stereo cameras from Point Grey Research. These stereo cameras are located on the sensor mast of the robot looking outward. The field-of-view of each camera is about 100 degrees, and the baseline is 12 cm; the height above ground is about 0.5 m, and the cameras are pointed forward at a slight angle. This arrangement presents a challenging situation: wide FOV and short baseline make distance errors large, and a small offset from the ground plane makes it difficult to track points over longer distances. We have developed a robust visual odometry solution that functions well under these conditions; we describe it in some detail in section 2.

A pair of wheel encoders and an Xsens IMU are used to complement the visual pose system. The IMU is located within the sensor mast and the wheel encoders are located on each of the front differential drive wheels. A Garmin GPS sensor is located on top of the sensor mast. This is a relatively cheap GPS sensor ($<$ 200 \$) and typical $2D$ error is 3-5 meters when the GPS receiver has 3D position fix (typically, when there's a good lock on 4 or more satellites). Errors in height, however, are much more uncertain. The GPS receiver provides position and velocity readings at 1Hz.

The 18th International Conference on Pattern Recognition (ICPR'06)
0-7695-2521-0/06 $20.00 © 2006 IEEE

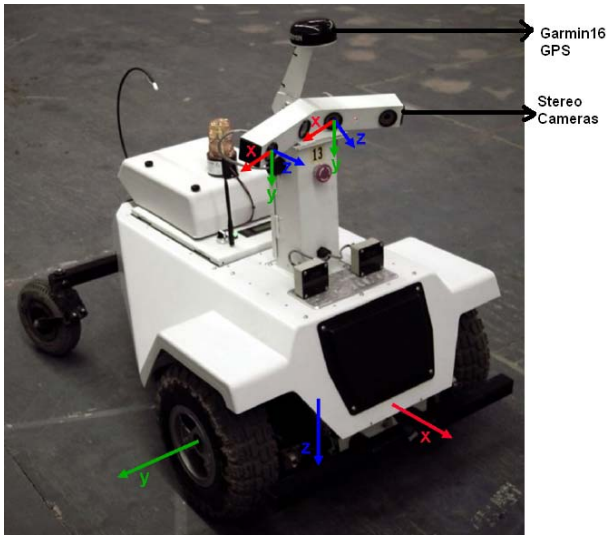



**Figure 1. Stereo and GPS Sensors on our Robot**

### 1.1 System Overview

Our visual odometry system uses feature tracks to estimate the relative motion between two frames. Corner feature points are detected in the left image of each stereo pair and tracked across frames. These feature points are then triangulated at each frame based on stereo correspondences. Three of these points are used to estimate the motion using absolute orientation. This motion is then scored using the pixel reprojection errors in both the cameras. We use the disparity space homography [1] to evaluate the inliers for the motion. In the end, the hypothesis with the best score (maximum number of inliers) is used as the starting point for a nonlinear minimization problem that minimizes the pixel reprojection errors in both the cameras simultaneously.

The relative motion between consecutive frames are chained together to obtain the absolute pose at each frame. The initial pose is obtained from the IMU, wheel encoders and GPS sensors by moving the robot in a straight line at speeds greater than 1 m/s. The IMU and the wheel encoders are also used to fill in the relative poses when visual odometry fails. Thus it complements the visual pose system. A very simple Kalman Filter is used to fuse global location and heading measurements from the GPS sensor, thereby avoiding long-term drift.

### 1.2 Related Work

Motion estimation from video is a well-studied problem in computer vision. Approaches for motion estimation are based on either dense optical flow or sparse feature tracks. [3] presents a visual odometry system that uses optical flow and a planar world assumption to obtain relative poses from a monocular camera. [13] uses optical flow for stereo images to compute the egomotion. Commonly used features for feature-based approaches are the Harris corners [8] or the more stable SIFT features [11]. [4] and [14] uses these corner tracks for monocular cameras. [19] uses the SIFT features for stereo egomotion estimation.

When compared to monocular video, motion estimation from stereo images is relatively easy and tends to be more stable and well behaved. Another major advantage of using stereo cameras is that one need not worry about the scale ambiguity present in monocular camera case. Approaches for binocular motion estimation [17] typically involve establishing feature correspondences. These feature points are triangulated and then an absolute orientation step is used to estimate the $3D$ motion. The use of $3D$ point correspondences to obtain the motion suffers from a major drawback – triangulations are much more uncertain in the depth direction. Therefore, these $3D$ points have non isotropic noise, and a $3D$ alignment between small sets of such $3D$ points gives poor motion estimates. To take into account this anisotropic noise in the $3D$ coordinates, Matei and Meer [12] presented an approach based on a technique from statistics called *bootstrap* to estimate the covariance for the $3D$ points and solve a *heteroscedastic, multivariate errors in variables* regression problem.

More recently, Nister et al; [15] described a visual odometry system. Their stereo algorithm proceeds by triangulating the feature points and then tracking them over time. The 3-point algorithm for single camera pose is then used to estimate the motion of the left camera. Each triplet of triangulated 3D points is used to generate a hypothesis in the RANSAC [6] framework. This hypothesis is then scored using pixel reprojections in both the left and the right cameras. In order to avoid drifting, the feature points are retriangulated often. Their system is efficient and can robustly estimate the motion over a few hundred meters with a stereo baseline of 28 cm. The key limitation of this approach is the fact that the hypothesis generation involves only one of the cameras and therefore is non-symmetrical. Our system, on the other hand, avoids this problem by using both the cameras to generate the hypothesis.

Similar to inertial odometry, pose obtained from visual odometry tends to accumulate error over time, thereby resulting in long-term drift without limit. To limit this drift, it is necessary to augment such local pose systems with global systems. GPS is an ideal companion for this purpose because it has a finite drift. It is a common practice to fuse these two sensing modalities through Extended Kalman Filtering [2, 10, 16].

The rest of the paper is organized as follows. Section 2

presents our visual odometry system. Section 3 describes our simple Kalman Filter to fuse GPS. Section 4 presents experimental results for outdoor navigation and finally section 5 concludes our presentation.

## 2 Motion Estimation from Images

### 2.1 Motion in Disparity Space

The disparity space [5, 1] is a projective space with isotropic noise that can be used for efficiently estimating the motion of a calibrated stereo rig. Consider this fixed stereo rig observing a moving rigid $3D$ scene. A point $M \equiv (X, Y, Z)^T$ undergoes a rigid motion with rotation $R$ and translation $t$ so that its new location is $M' \equiv (X', Y', Z')^T$. The point $M$ projects in the left image to the point $(x, y)$ and its disparity is $d$. Let $\omega \equiv (x, y, d)^T$ and $\omega' \equiv (x', y', d')^T$ correspond to $M$ and $M'$ respectively in the disparity space. Then, it can be shown [5]

$$\begin{pmatrix} \omega' \\ 1 \end{pmatrix} \simeq H(R,t) \begin{pmatrix} \omega \\ 1 \end{pmatrix} \quad (1)$$

Thus, in the disparity space, a $3D$ point undergoing a rigid euclidean transformation transforms according to the homography $H(R, t)$. Given the euclidean motion and the camera parameters, it is straightforward to deduce this homography $H$ and vice-versa.

### 2.2 Feature Detection and Tracking

Harris corners [8] are detected in the left and the right image of each frame in the video sequence. The features detected in the left image are matched to features in the right image of the same row using normalized cross correction (NCC) over a $11 \times 11$ window. Similarly, the features in the right image are matched to the left image. Only those features that are matched to each other and with a NCC value above a threshold (taken to be 0.5) are retained. This makes the scheme more robust as only features that achieve the lowest NCC scores in each other are reliable. These features are also matched with the features from the subsequent frame to give feature tracks. Since we have continuous video, a feature point can move only a fixed maximum distance between consecutive frames. For each feature point in the current frame, its NCC is evaluated for every feature point in the next frame that lies within a specified distance of its location in the current frame. This distance is taken to be 50 for our setup. As in the stereo matching step, those pixels that achieve mutual minimum NCC are retained and defined as "matched".

### 2.3 Motion Estimation

Starting with these potential matches, our RANSAC-based motion estimation involves

**Hypothesis generation** Three points are required to generate a motion hypothesis. To get reliable motion, we must ensure that these three points are spaced out well in the image. Points that are too close in the image are unlikely to give good estimates of the motion. Therefore, for any selection of three points, we check to see if they are sufficiently spread out in the image. This is accomplished by dividing the feature $xy$ locations in the image into equally spaced bins and selecting each feature point from a different bin. The bin size in our case is taken to be 32.

Next, we triangulate these three points to obtain their $3D$ locations $M_i$ and $M_i'$. We then seek the rotation matrix $R$ and the translation $t$ such that $M_i' = RM_i + t$. This is a standard absolute orientation problem and can be solved efficiently using singular value decomposition [23]. Most of the computations above involve operations on $3 \times 3$ matrices and require few flops. The most time-consuming operation here is the computation of SVD of a 3 matrix. We obtain a closed form expression for the SVD of this matrix to save computational time, resulting in a very fast implementation.

**Hypothesis scoring** Corresponding to each rotation and translation pair hypothesis $(R, t)$, the disparity space homography $H(R, t)$ can be calculated using equation 1. For a hypothesized correspondence in the disparity space $\omega_i \leftrightarrow \omega_i'$, the homography is applied to $\omega_i$, resulting in the point $\omega_i''$. The reprojection error is then given by $\varepsilon_i = |\omega_i' - \omega_i''|$.

A correspondence is taken as an inlier to this homography if the inifinity norm of the error vector $|\varepsilon_i|_\infty$ is less than a predefined maximum threshold value. In our implementation, we have taken this threshold value to be $1.25$ pixels. The number of inlier matches to a motion is taken as its score. Since each of the hypotheses generated during the RANSAC needs to be scored with all the feature correspondences, it is extremely important to code this efficiently. We have coded these routines using SIMD instructions.

**Nonlinear minimization** The RANSAC is applied for a fixed number of samples (in our case a maximum of 500 samples are taken). The hypothesis with the best score (maximum number of inliers) is used as the starting point for a nonlinear minimization algorithm. We use the Levenberg-Marquardt algorithm for nonlinear least squares minimization. The Jacobian required for this minimization is approximated by forward differencing. Since the $3 \times 3$ rotation matrix has only three degrees of freedom, we work with the euler angles instead. The variables for this minimization are the three euler angles and the three translation parameters.

For $N$ matches, $\omega_i \leftrightarrow \omega_i'$, $i = 1, \ldots, N$, the error function to be minimized is given by $\min \sum_{i=1}^{N} \|\omega_i'' - \omega_i'\|^2$

The starting point for this nonlinear minimization routine is very good and hence the procedure converges to a local minima within only 5 to 10 iterations. We have observed that this nonlinear minimization step makes a significant difference to the computed motion. This minimization step gives us the full six-degrees-of-freedom pose constraint between the two frames. The covariance estimate of the pose is approximated from the Jacobian $J$ of the error function as $C = (J^t J)^{-1}$.

We have found that the approach outlined above is very efficient ($> 15$Hz) and works remarkably well, even for stereo rigs with a small baseline. The fact that we are triangulating the feature points for each frame, builds a firewall for error propagation. However, this also means that there will be a drift when the rig is stationary. In order to avoid this drift, we update the reference frame (the frame with reference to which the motion of the next frame is computed) only when the robot has moved some minimum distance (taken to be 5 cm in our implementation). Since, we are re-triangulating for every frame, it is important to calibrate the stereo cameras well. A standard plane based calibration step works well for all our experiments.

The fundamental reason that our approach gives reliable motion estimates, even in small-baseline situations is due to the fact that we stick to image-based quantities and use both the left and right images symmetrically. The absolute orientation step used to generate the hypothesis uses the left and the right cameras symmetrically to generate the motion hypothesis. The hypothesis is evaluated and scored based on reprojection errors in both views, resulting in an accurate estimate of the motion. This estimate is then refined in the nonlinear minimization step which also uses the two cameras uniformly.

## 3 Local and Global Consistency

The errors in visual odometry(VO) and the errors in GPS are in some respects complementary – VO provides accurate locally-consistent information about pose, which is unbounded over long term. GPS, when available, has bounded error in the X,Y direction of about 3 meters standard deviation under good conditions, but is subject to small jumps and drifts over the short term. It is difficult to filter these errors correctly using only IMU and wheel data, since the robot could be moving contrary to the wheel odometry, and the IMU position estimates drift rapidly.

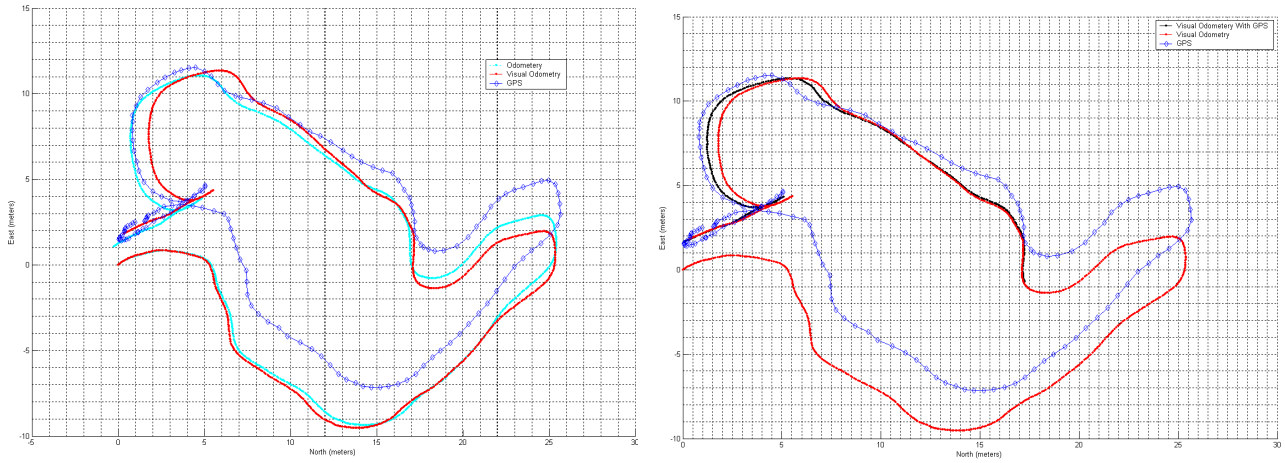
In order to maintain global consistency, the VO pose can be modified using GPS data in a Kalman filter. Since GPS is unreliable in height measurements, the Kalman Filter state is taken to be the North and East position, together with the heading. The GPS provides the necessary measurements for the Kalman Filter to correct the pose estimates. Position information is applied to correct the position states and the velocity information is independently used to correct the heading. Position information is used when the GPS receiver has at least a 3D position fix and the velocity information is only used when the vehicle is travelling 0.5 m/s or faster, to limit the effect of velocity noise from GPS on the heading estimate. In addition, GPS measurements are used only if the robot has travelled a certain distance from the last GPS measurement. This will ensure that the robot's pose does not change due to GPS jumps when the vehicle is stationary. The filter essentially nudges the VO pose towards global consistency, while maintaining accurate local consistency. Over larger loops, of course, the 3 m deviation of the GPS unit means that the map may not be consistent. In this case, other techniques such as wide-baseline image matching [11] would have to be employed.

## 4 Experimental Results

We have implemented and tested our integrated pose system on several outdoor terrains. Figure 3 shows a typical outdoor image captured by the left camera of the stereo pair. Since GPS is accurate to only about 3-4 meters, in order to validate our results, the robot was moved in a closed loop over 50 - 100 meters. Since the starting and the ending point are the same, the difference in pose between these two points gives a good indication of the error in localization. We measure this error in percentage over the total distance.

**Figure 3. Typical outdoor terrain as seen from the left camera**

Table 1 compares this error for the raw IMU/wheels pose, visual odometry and the GPS integrated visual odometry for four loops. Except for the first loop, the visual odometry outperforms the raw vehicle odometry significantly. The integration of the visual odometry with GPS also substantially reduces the loop closure error. Loop 2 is




(a) Raw odometry compared to raw visual odometry and GPS

(b) Visual odometry integrated with GPS

**Figure 2. Comparison of Odometry, Visual Odometry and GPS for loop 1**

the longest and covers 141 meters with the average speed of the robot being 0.7 m/s and involves many turns. For this case, the visual odometry outperforms vehicle odometry by a factor of two. Furthermore, long term drifts are compensated for by the integration of GPS and visual odometry resulting in a loop closure error of less than 0.5 m. For loop 1, the vehicle odometry performs marginally better than visual odometry. This can be explained by the fact that the robot was moved at a slow speed (average speed about 0.5 m/s) and this loop did not involve any sharp turns. Thus allowing the raw vehicle odometry to do a good job. However, as the robot speed increases or the turns become sharper or more frequent, the wheel slippage increases and the performance of vehicle odometry degrades as is evident in the other three loops. Figure 2 compares the robot pose as computed from raw vehicle odometry, visual odometry, raw GPS and the GPS integrated visual odometry for loop 1.

The wheel slippage is especially prominent in loop 4, where the robot was slipping in mud for a substantial amount of time. Figure 4 plot the poses for this case. The wheel slippage is marked in figure 4(a). Since the wheels are turning, the vehicle odometry fails to detect that the robot is stationary, resulting in a substantial drift. However, it is easy to detect no motion for visual odometry since the image features do not move much. Thereby, the visual odometry pose remains correct and localizes the robot well.

**Table 1. Loop Closure Error in Percentage**

| Run Number | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| Distance(meters) | 82.4 | 141.6 | 55.3 | 51.0 |
| Method | Percentage Error | | | |
| Vehicle Odometry | 1.3 | 11.4 | 11.0 | 31.0 |
| Raw Visual Odometry | 2.2 | 4.8 | 5.0 | 3.9 |
| Visual Odometry & GPS | 2.0 | 0.3 | 1.7 | 0.9 |

## 5 Conclusion

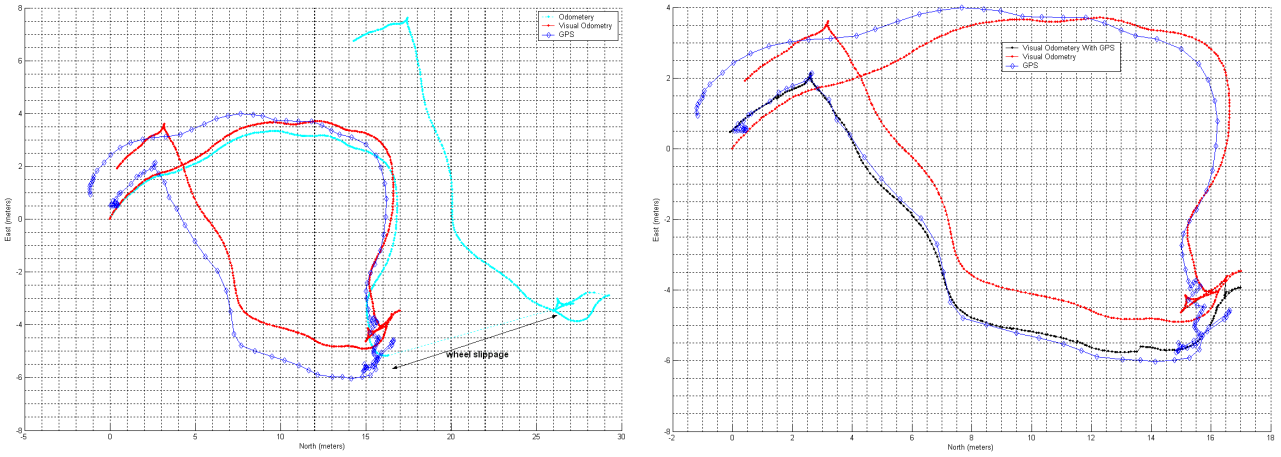
We have presented a real-time system for robot localization in outdoor navigation tasks using stereo vision. Our visual odometry system is robust, reliable and accurate with errors on the order of a few meters over hundred meters or more. This when integrated with an inexpensive GPS prevents long-term drifts allowing the robot to stay localized over long distances.

Although, motion estimation from video has been a widely researched topic in computer vision, real-time systems utilizing vision for localization of robots have been very few. We hope to have demonstrated stereo vision as a powerful sensor for localization of mobile robots in outdoor environments. As cameras become less expensive, vision-based localization will provide a cost-effective odometry system that can be used to complement traditional inertial odometry systems to provide accurate localization in outdoor navigation tasks.

## References

[1] M. Agrawal, K. Konolige, and L. Iocchi. Real-time detection of independent motion using stereo. In *IEEE workshop on Motion (WACV/MOTION)*, January 2005.

[2] D. Bouvet, M. Froumentin, and G. Garcia. A real-time localization system for compactors. *Automation in Construction*, 10:417–428, 2001.




(a) Raw odometry compared to raw visual odometry and GPS

(b) Visual odometry integrated with GPS

**Figure 4. Comparison of Odometry, Visual Odometry and GPS for loop 4: Case of spinning wheels**


[3] J. Campbell, R. Sukthankar, and I. Nourbakhsh. Techniques for evaluating optical flow for visual odometry in extreme terrain. In *Proceedings of International Robotics Symposium*, October 2004.

[4] A. Davison. Real-time simultaneaous localisation and mapping with a single camera. In *Proc. International Conference on Computer Vision (ICCV)*, pages 1403–1410, 2003.

[5] D. Demirdjian and T. Darrell. Motion estimation from disparity images. In *Proc. International Conference on Computer Vision*, volume 1, pages 213–218, July 2001.

[6] M. Fischler and R. Bolles. Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography. *Commun. ACM.*, 24:381–395, 1981.

[7] H. H. Gonzalez-Banos and J. C. Latombe. Navigation strategies for exploring indoor environments. *International Journal of Robotics Research*, 21(10-11):829–848, Oct-Nov 2002.

[8] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, pages 147–151, 1988.

[9] K. Konolige. Large-scale map-making. In *Proceedings of the National Conference on AI (AAAI)*, 2004.

[10] P. Lamon and R. Siegwart. 3d-odometry for rough terrain – towards real 3d navigation. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, May 2003.

[11] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[12] B. Matei and P. Meer. Optimal rigid motion estimation and performance evaluation with bootstrap. In *Proc. Computer Vision and Pattern Recognition Conference*, pages 339–345, 1999.

[13] L.-P. Morency and R. Gupta. Robust real-time egomotion from stereo images. In *Proc. International Conference on Image Processing*, 2003.

[14] D. Nister. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 26(6):756–770, June 2004.

[15] D. Nister, O. Naroditsky, and J. Bergen. Visual odometry. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, June 2004.

[16] K. Ohno, T. Tsubouchi, B. Shigematsu, and S. Yuta. Differential gps and odometry-based outdoor navigation of a mobile robot. *Advanced Robotics*, 18(6):611–635, 2004.

[17] C. F. Olson, L. H. Matthies, M. Schoppers, and M. W. Maimone. Robust stereo ego-motion for long distance navigation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 453–458, 2000.

[18] E. Royer, M. Lhuillier, M. Dhome, and T. Chateau. Localization in urban environments: monocular vision compared to a differential gps sensor. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2005.

[19] S. Se, D. Lowe, and J. Little. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *International Journal of Robotic Research*, 21:735–758, August 2002.

[20] J. D. Tards, J. Neira, P. M. Newman, and J. J. Leonard. Robust mapping and localization in indoor environments using sonar data. *International Journal of Robotics Research*, 21(4):311–330, 2002.

[21] S. Thrun. Robotic mapping: A survey. In G. Lakemeyer and B. Nebel, editors, *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann, 2002.

[22] S. Thrun, D. Hahnel, D. Ferguson, M. Montemerlo, R. Triebel, W. Burgard, C. Baker, Z. Omohundro, S. Thayer, and W. Whittaker. A system for volumetric robotic mapping of abandoned mines. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2003.

[23] S. Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(4), April 1991.