Jacques Blanc-Talon
Wilfried Philips
Dan Popescu
Paul Scheunders (Eds.)

Advanced Concepts for Intelligent Vision Systems

LNCS 4678

9th International Conference, ACIVS 2007
Delft, The Netherlands, August 2007
Proceedings

Springer

Image Interpretation

Image Coding

An Efficient Closed-Form Solution to Probabilistic 6I Visual Odometry for a Stereo Camera

F.A. Moreno, J.L. Blanco, and J. González

Department of System Engineering and Automation, University of Málaga, Spain
famoreno@isa.uma.es, {jlblanco,jgonzalez}@ctima.uma.es

Abstract. Estimating the ego-motion of a mobile robot has been traditionally achieved by means of encoder-based odometry. However, this method presents several drawbacks, such as the existence of accumulative drifts, its sensibility to slippage, and its limitation to planar environments. In this work we present an alternative method for estimating the incremental change in the robot pose from images taken by a stereo camera. In contrast to most previous approaches for 6D visual odometry, based on iterative, approximate methods, we propose here to employ an optimal closed-form formulation which is more accurate, efficient, and does not exhibit convergence problems. We also derive the expression for the covariance associated to this estimation, which enables the integration of our approach into vision-based SLAM frameworks. Additionally, our proposal combines highly-distinctive SIFT descriptors with the fast KLT feature tracker, thus achieving robust and efficient execution in real-time. To validate our research we provide experimental results for a real robot.

1 Introduction

Odometry is one of the most widely used means for estimating the motion of a mol robot. Traditionally, odometry is derived from encoders measuring the revolutions the robot's wheels, thus providing information for estimating the change in the ro pose. Unfortunately, the usage of encoder-based odometry is limited to wheeled bots operating on plane surfaces and systematic errors such as drift, wheel slippa and un-controlled differences in the robot's wheels induce incremental errors in displacement estimation, which can not be properly modelled by a zero-mean Ga sian distribution. This erroneous assumption about the encoder-based odometry err is accepted in most probabilistic filters for robot localization and SLAM [15], a may eventually lead to the divergence of the filter estimation.

In order to overcome the limitations of encoder-based odometry, other n proprioceptive sensors such as laser sensors [4, 14] and, more recently, vision-ba systems [1, 16] have been used in the last years. The proper performance of laser s sors is also limited to purely planar motions, whereas vision-based odometry explo the advantages of the wider field-of-view of cameras. Nowadays, cameras are che and ubiquitous sensors capable of collecting huge amount of information from the vironment. The existence of powerful methods for extracting and tracking signific features from images, along with the above-mentioned advantages of cameras, esta lish a propitious framework for applying vision to ego-motion estimation.

J. Blanc-Talon et al. (Eds.): ACIVS 2007, LNCS 4678, pp. 932–942, 2007.

Regarding this topic, several approaches have been proposed in the technical literature which apply different methods for estimating the displacement of a vision-equipped mobile robot from a sequence of images taken along its navigation through the environment. The work in [10] reports both a monocular and a stereo visual odometry system based on iterative methods for estimating the 3D change in robot pose, while [1] performs monocular visual odometry with uncalibrated consumer-grade cameras under the assumption of purely planar motion. In [13] it is presented a probabilistic method for performing SLAM which uses visual odometry as the robot motion model. This approach looks for sets of features in the stereo images and computes their SIFT descriptors in order to establish correspondences. The camera motion is subsequently estimated using an iterative optimization algorithm which minimizes the re-projection error of the 3D points.

In this paper we propose a new approach to visual odometry by estimating incremental changes in the 6D (*yaw*, *pitch*, *roll*, x, y, z) robot pose between consecutive stereo images. Our method estimates the complete set of angles and translations, thus there are not constraints about the potential movements of the camera as in other approaches like [2].

Our algorithm combines the speed of the Kanade-Lucas-Tomasi detector and tracker [12] with the selectivity of SIFT descriptors [8] to match features in the stereo images. Since SIFT-based stereo matching is only carried out when the number of distinctive points in the tracker falls below a given threshold, we avoid the high computational cost involved in computing and comparing the Euclidean distance between SIFT descriptors for all the features in each pair of stereo images. Another advantage of our approach over previous works is the application of a closed form solution to estimate the changes in orientation and translation, eluding both the complexity and divergence problems of iterative methods. Moreover, we model the uncertainty of the pose estimate by propagating the uncertainty in the 3D positions of the observed points.

The rest of the paper is organized as follows: Section II presents a brief outline of our proposed method for performing visual odometry, which is described in more detail in section III. In section IV we provide some experimental results, whereas section V presents some conclusions and the future work.

2 Method Overview

Our proposed method, depicted in Fig. 1, can be summarized by the following stages:

1. Searching for a set of interest features in a first pair of stereo images, and computation of their corresponding SIFT descriptors.
2. Stereo matching based on the Euclidean distance between descriptors and epipolar geometry restrictions.
3. Projection into 3D space of the matched features, therefore obtaining a set of three dimensional points with coordinates relative to the current robot pose.
4. Tracking the features in the next pair of stereo images. Notice that this tracking allows us to avoid a new SIFT-based matching step.
5. These tracked features are projected into 3D space, yielding a new set of three dimensional points with known correspondences to the previous set of 3D points.

6. Robot (camera) 6D pose estimation through a closed-form solution of the absolute orientation problem [6], given the correspondences between the two sets of 3D points.
7. If the number of tracked features falls below a certain threshold, new features are searched in the stereo images and their SIFT descriptors computed. Subsequently, they are matched according to their descriptors and added to the current set of points.
8. Repeat from step 4.

A full detailed description of all the steps of our method is presented in next section.

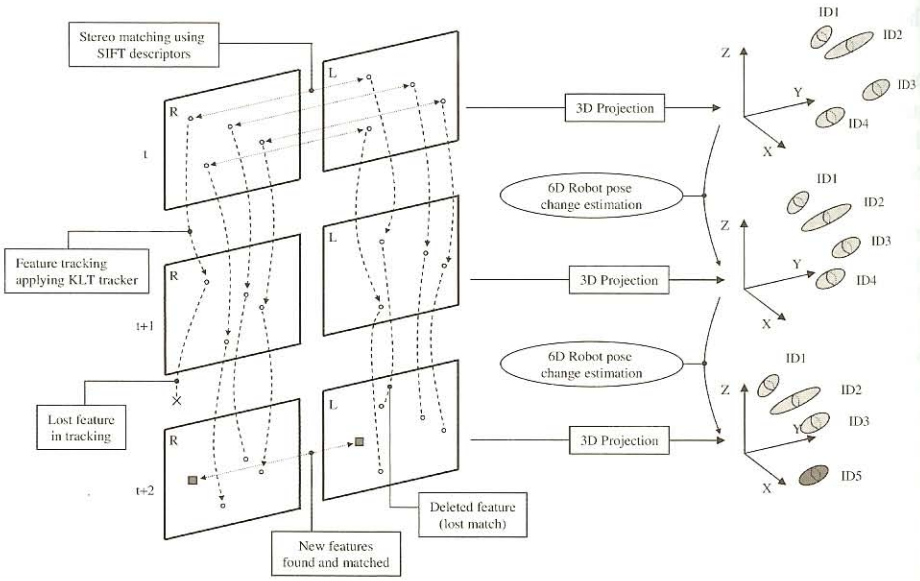

Fig. 1. A schematic representation of the proposed method

3 Detailed Description of the Method

This section presents a detailed description of the different operations involved in our proposed algorithm for performing visual odometry.

3.1 Extraction and Matching of Reliable Features from Stereo Images

Several methods have been proposed in the literature for extracting interest points from images, as the well known detectors of Kitchen & Rosenfeld [7] and Harris [5], based on the first and the second-order derivatives of images, respectively. More recently, the SIFT detector proposed by Lowe [8] deals with this problem by identifying local extrema in a pyramid of Difference of Gaussians (DoG). It also provides the detected features with a descriptor that exhibits invariance to rotation and scale, and partial invariance to lighting changes and affine distortions. In our work, the detection

of interest points in the images is carried out by the method proposed by Shi and Tomasi [12]. In addition, their corresponding SIFT descriptors are also computed to make them sufficiently distinguishable and to improve the robustness of the matching process.

Once a set of keypoints has been detected in the left and right images they are robustly matched according to both the similarity of their descriptors and the restrictions imposed by the epipolar geometry. More precisely, in the former restriction, for each keypoint in the left image it is computed the Euclidean distance between its descriptor and those of the keypoints in the right image. For a pair of keypoints to be considered as a candidate match their descriptors must fulfill two conditions: to be similar enough (their distance below a certain threshold), and different enough to other candidates (their distance above a certain threshold). Moreover, the points must fulfill the epipolar constraint, i.e. they have to lay on the conjugate epipolar lines (or be close enough). In a stereo vision system with parallel optical axis as the one we use here, the epipolar lines are parallel and horizontal, thus the epipolar constraint reduces to checking that both features are in the same row.

Finally, each pair of matched features is assigned a unique ID which will be used to identify the point projected from their image coordinates in subsequent time steps.

3.2 Projection into 3D Space

Once the features have been robustly matched, the coordinates of their corresponding 3D points are estimated from their coordinates on the images and the intrinsic parameters of the stereo system. Formally, let (c, r) be the image coordinates of a feature in the left image (which we will be taken as the reference one) and d the disparity of its conjugate feature in the right one. Then, the 3D coordinates (X,Y,Z) of the projected point are computed as:

$$X = (c - c_0)\frac{b}{d} \quad Y = (r - r_0)\frac{b}{d} \quad Z = f\frac{b}{d} \tag{1}$$

where (c_0, r_0) are the image coordinates of the principal point in the reference image, b is the baseline of the stereo system, and f stands for the identical focal length of the cameras.

The errors in the so obtained variables r, c, and d are usually modeled as uncorrelated zero-mean Gaussian noises [9]. By using a first-order error to approximate the distribution of the variables in (1) as multivariate Gaussians, we obtain the following covariance matrix:

$$\Sigma = \begin{pmatrix} \sigma_X^2 & \sigma_{XY} & \sigma_{XZ} \\ \sigma_{XY} & \sigma_Y^2 & \sigma_{YZ} \\ \sigma_{XZ} & \sigma_{YZ} & \sigma_Z^2 \end{pmatrix} = \mathbf{J}\, diag\left(\sigma_c^2, \sigma_r^2, \sigma_d^2\right)\mathbf{J}^{\mathrm{T}} \tag{2}$$

where $\mathbf{J}$ stands for the Jacobian matrix of the functions in (1), and $\sigma_X^2, \sigma_Y^2, \sigma_Z^2, \sigma_c^2, \sigma_r^2$, and σ_d^2 are the variances of the corresponding variables. Expanding (2) we come up with the following expression for Σ:

$$\Sigma = \begin{pmatrix} \frac{b^2\sigma_r^2}{d^2} + \frac{b^2(c-c_0)^2\sigma_d^2}{d^4} & \frac{(c-c_0)b^2\sigma_d^2(r-r_0)}{d^4} & \frac{(c-c_0)b^2\sigma_d^2 f}{d^4} \\ \frac{(c-c_0)b^2\sigma_d^2(r-r_0)}{d^4} & \frac{b^2\sigma_r^2}{d^2} + \frac{b^2(r-r_0)^2\sigma_d^2}{d^4} & \frac{(r-r_0)b^2\sigma_d^2 f}{d^4} \\ \frac{(c-c_0)b^2\sigma_d^2 f}{d^4} & \frac{(r-r_0)b^2\sigma_d^2 f}{d^4} & \frac{f^2b^2\sigma_d^2}{d^4} \end{pmatrix} \quad (3)$$

which approximately models the uncertainty in the 3D coordinates of points computed from the noisy measurements of a stereo system.

Finally, to distinguish it from the rest of the projected points, each 3D point is assigned the unique ID of the matched pair of image features from which it was generated.

3.3 Tracking Features

In successive stereo frames, the detected features are tracked using the well-known Lucas-Kanade-Tomasi method [12] in order to determine their coordinates in the new pair of stereo images. This method computes the optical flow of a pixel in two consecutive images by minimizing the difference between the surrounding windows using a Newton-Raphson method.

The correct tracking of a pair of matched features in the left and right images at time k yields another matched pair of features in the stereo images at time k+1. At this point the epipolar constraint is considered to detect improperly tracked features and, hence, to avoid the presence of unreliable matched pairs. By using this tracking process, we avoid both the search for features and the SIFT-based stereo matching at the new camera pose. Thus this method speeds up the process of extracting and matching features and, consequently, the computational burden of the whole visual odometry procedure is considerably reduced.

The resulting set of tracked features are also projected to space following the method described in section 3.2, yielding a new set of 3D points which keep their IDs from the image features in order to maintain an implicit matching relationship with the points in the previous set.

If the number of tracked features falls below a threshold, the algorithm searches for new features in the images to maintain a proper amount of elements in the 3D point sets.

3.4 Probabilistic Estimation of the Pose Change

In this section we present a method for estimating the probability distribution of the change in the robot pose between two time steps from the sets of 3D points determined as described above.

Formally, let $\mathbf{X}_k$ be a set of 3D points obtained at time k:

$$\mathbf{X}_k = \left\{ \mathbf{X}_k^i \right\}_{i=1...N_1} \quad (4)$$

where the position of each 3D point $\mathbf{X}_k^i$ is assumed to follow a Gaussian distribution with mean $\mu_{X_k^i} = \left\langle X_k^i, Y_k^i, Z_k^i \right\rangle$ and covariance $\Sigma_{\mathbf{X}_k^i}$ determined by equations (1) and (3), respectively:

$$\mathbf{X}_k^i \sim N\left(\mu_{\mathbf{X}_k^i}, \Sigma_{\mathbf{X}_k^i}\right) \tag{5}$$

At this point, we define $q_{k,k+1}$ as the random variable which models the pose change between time steps k and k+1 as a function of the sets of projected 3D points $\mathbf{X}_k$ and $\mathbf{X}_{k+1}$:

$$q_{k,k+1} = f\left(\mathbf{X}_k, \mathbf{X}_{k+1}\right); \qquad q_{k,k+1} \sim N(\mu_q, \Sigma_q) \tag{6}$$

Under a linear approximation of error propagation, $q_{k,k+1}$ follows a Gaussian distribution with covariance matrix Σ_q and mean $\mu_q = \left\langle \Delta x, \Delta y, \Delta z, \Delta\alpha, \Delta\beta, \Delta\gamma \right\rangle$ where Δx, Δy, and Δz are the increments in the X, Y, and Z coordinates respectively, and $\Delta\alpha$, $\Delta\beta$, and $\Delta\gamma$ stand for the increments in the yaw, pitch, and roll angles, respectively.

3.4.1 Estimation of the Mean Value μ_q

In this paper, we propose to compute μ_q through the method reported by Horn in [6], where it is derived a closed-form solution to the least-squares problem of finding the relationship between two coordinate systems using the measurements of the coordinates of a number of points in both systems. We use the mean values $\mu_{X_k^i}$ of the positions of the 3D points as the inputs to this algorithm. This closed-form solution is in contrast to other proposals for visual odometry based on iterative methods [11, 13] which require an initial estimation of the change in pose.

The closed-form solution can be summarized as follows:

1. Compute the centroids ($\mathbf{c}_k$ and $\mathbf{c}_{k+1}$) of the two sets of points and subtract them from their coordinates in order to deal only with coordinates relative to their centroids: $\overline{\mathbf{X}}_k^i = \left\langle \overline{X}_k^i, \overline{Y}_k^i, \overline{Z}_k^i \right\rangle$ and $\overline{\mathbf{X}}_{k+1}^i = \left\langle \overline{X}_{k+1}^i, \overline{Y}_{k+1}^i, \overline{Z}_{k+1}^i \right\rangle$.
2. For the i-th 3D point, compute the following nine products of its coordinates at time k and k+1:

$$P_{XX}^i = \overline{X}_k^i \overline{X}_{k+1}^i \quad P_{XY}^i = \overline{X}_k^i \overline{Y}_{k+1}^i \quad \ldots \quad P_{ZY}^i = \overline{Z}_k^i \overline{Y}_{k+1}^i \quad P_{ZZ}^i = \overline{Z}_k^i \overline{Z}_{k+1}^i \tag{7}$$

3. Accumulate the products in (7) for all the 3D points to end up with the following nine values:

$$S_{XX} = \sum_i P_{XX}^i \quad S_{XY} = \sum_i P_{XY}^i \quad \ldots \quad S_{ZY} = \sum_i P_{ZY}^i \quad S_{ZZ} = \sum_i P_{ZZ}^i \tag{8}$$

4. Form a 4x4 symmetric matrix with the elements in (8):

$$\mathbf{N} = \begin{bmatrix} S_{XX}+S_{YY}+S_{ZZ} & S_{YZ}-S_{ZY} & S_{ZX}-S_{XZ} & S_{XY}-S_{YX} \\ S_{YZ}-S_{ZY} & S_{XX}-S_{YY}-S_{ZZ} & S_{XY}+S_{YX} & S_{ZX}+S_{XZ} \\ S_{ZX}-S_{XZ} & S_{XY}+S_{YX} & -S_{XX}+S_{YY}-S_{ZZ} & S_{YZ}+S_{ZY} \\ S_{XY}-S_{YX} & S_{ZX}+S_{XZ} & S_{YZ}+S_{ZY} & -S_{XX}-S_{YY}+S_{ZZ} \end{bmatrix} \tag{9}$$

5. Find the eigenvector corresponding to the largest eigenvalue of $\mathbf{N}$, which will be taken as the quaternion that determines the rotation between the robot pose at time steps k and k+1.
6. Compute the rotation matrix ($\mathbf{R}$) associated to the so obtained quaternion, and compute the translation $\mathbf{t} = (\Delta x, \Delta y, \Delta z)^{\mathrm{T}}$ as the difference between the centroid at time k and the scaled and rotated centroid at time k+1:

$$\mathbf{t} = \mathbf{c}_k - \mathbf{R}\mathbf{c}_{k+1} \tag{10}$$

Finally, we extract the values of the increments in the yaw, pitch, and roll angles $\langle \Delta\alpha, \Delta\beta, \Delta\gamma \rangle$ between poses from the rotation matrix $\mathbf{R}$, having in this way all the components of μ_q.

3.4.2 Estimation of the Covariance Matrix Σ_q

Covariance matrixes are usually obtained through a linear approximation of the functions involved in a given transformation between variables (see, for example, section 3.2). However, in the case of the closed-form solution described above the function cannot be linearized due to the computation of the largest eigenvector.

Therefore, we propose here to use the linearized version of the problem, which can be stated as the minimization of the least square error of the system:

$$\begin{pmatrix} X^i_{k+1} \\ Y^i_{k+1} \\ Z^i_{k+1} \\ 1 \end{pmatrix} = \left(\begin{array}{ccc:c} & \mathbf{R} & & \mathbf{t} \\ \hdashline 0 & 0 & 0 & 1 \end{array}\right) \begin{pmatrix} X^i_k \\ Y^i_k \\ Z^i_k \\ 1 \end{pmatrix} \tag{11}$$

for the variables which determines the pose change, i.e. $\mu_q = \langle \Delta x, \Delta y, \Delta z, \Delta\alpha, \Delta\beta, \Delta\gamma \rangle$. Expanding (11) we obtain the position of the i-th point at time k+1 as a function of its position at time k (represented by $\mathbf{X}^i_k$) and the increments in X, Y, Z, yaw, pitch and roll between both time steps:

$$\begin{aligned} X^i_{k+1} &= f\left(\mu_q, \mathbf{X}^i_k\right) \\ Y^i_{k+1} &= f\left(\mu_q, \mathbf{X}^i_k\right) \\ Z^i_{k+1} &= f\left(\mu_q, \mathbf{X}^i_k\right) \end{aligned} \tag{12}$$

By linearizing these equations we come to the following expression for Σ_q:

$$\Sigma_q^{-1} = \mathbf{H}^{\mathrm{T}} \Sigma^{-1} \mathbf{H} = \sum_i \mathbf{H}^{i\mathrm{T}} \left(\Sigma^i\right)^{-1} \mathbf{H}^i \tag{13}$$

where $\mathbf{H}^i$ stands for the Jacobian matrix of the equations in (12) for the i-th 3D point relative to μ_q and $\Sigma^i = \Sigma^i_k + \Sigma^i_{k+1}$ is the sum of the position covariance matrices of the i-th point at times k and k+1 as defined in equation (3). Notice that, since the 3D points are uncorrelated, the first expression in (13) can be split into the sum of its block diagonal elements.

4 Experimental Results

We have performed a variety of experiments to compare classical encoder-based odometry with our proposed method for visual odometry in an indoor environment. In this paper, we present one of them where our robot Sancho is equipped with a PointGrey Bumblebee stereo camera and driven through a room while gathering stereo images and odometry readings. We also use laser scans to build a map of the environment and estimate the real path of the robot, which will be taken as the ground truth in this experiment (thick lines in Fig. 2(a)–(b)). An example of the scene managed in this experiment is shown in Fig. 2(c).

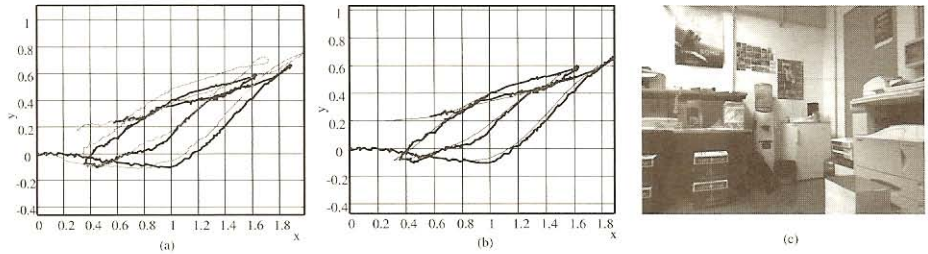

Fig. 2. (a) Path of the robot estimated from the laser scanner built map (thick line) and our proposed visual odometry method (thin line). (b) Estimated paths from the laser scanner map and the encoder-based odometry readings (dashed line). (c) Example of the images managed in the experiments.

In order to compare the performance of the odometry methods, we compute the errors committed by both methods at each time step as the difference between their estimates and the ground truth.

The histograms of the 3D position errors of both approaches are shown in Fig. 3. We have found that both methods perform similarly, with most of the errors in Δx and Δy below 5 cm. Notice that since the robot moves in a planar environment, Δz should be zero for the whole experiment. Consequently, our algorithm provides a coherent estimation which is always close to $\Delta z = 0$ with a small error (typically 1 cm), as can be seen in Fig.3. The distribution of the error in the 3D position is illustrated in the last plot in Fig. 4.

Regarding the estimation of the orientation, visual odometry achieves an error in *yaw* (the only rotational degree of freedom of a planar robot) similar to conventional odometry. However, we should highlight the accuracy of our algorithm in the other components of the orientation, where the largest error is below 1 deg (please, refer to the histograms for *pitch* and *roll* in Fig.4).

Recalling the estimated paths of the robot in Fig. 2 according to both odometric methods, we can now remark their similar accuracy in spite of the higher dimensionality of visual odometry, which, a priori, is prone to accumulate larger errors. We can

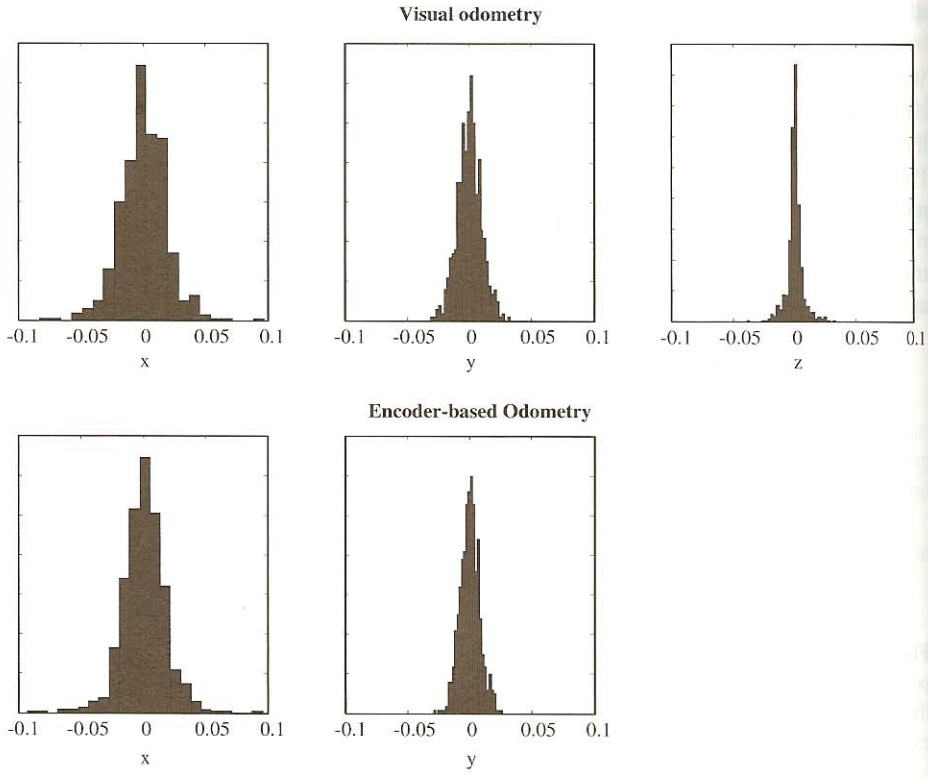

Fig. 3. Histograms of the errors committed in the estimation of the changes in the robot position for the visual odometry (top plots) and classical odometry (bottom plots) approaches

conclude that the reason for this performance is the small estimation errors of visual odometry in the dimensions not involved in planar odometry, i.e. Δz, $\Delta\beta$, $\Delta\gamma$.

5 Conclusions

This paper has presented a new method to perform visual odometry by computing the 6D change between the poses of a camera in consecutive time steps. Our method combines the speed of the Lucas-Kanade-Tomasi detector and tracker with the capability of the SIFT descriptor to distinguish features. Another contribution of this work in comparison to previous approaches is the employment of a closed-form, optimal solution to the problem of finding the 6D transformation between two sets of corresponding points. The results show that the performance of our approach for visual odometry is quite similar to that of conventional odometry for planar environments, whereas visual odometry additionally allows movements in 6D. Further research will be aimed at integrating the presented approach into visual SLAM frameworks.

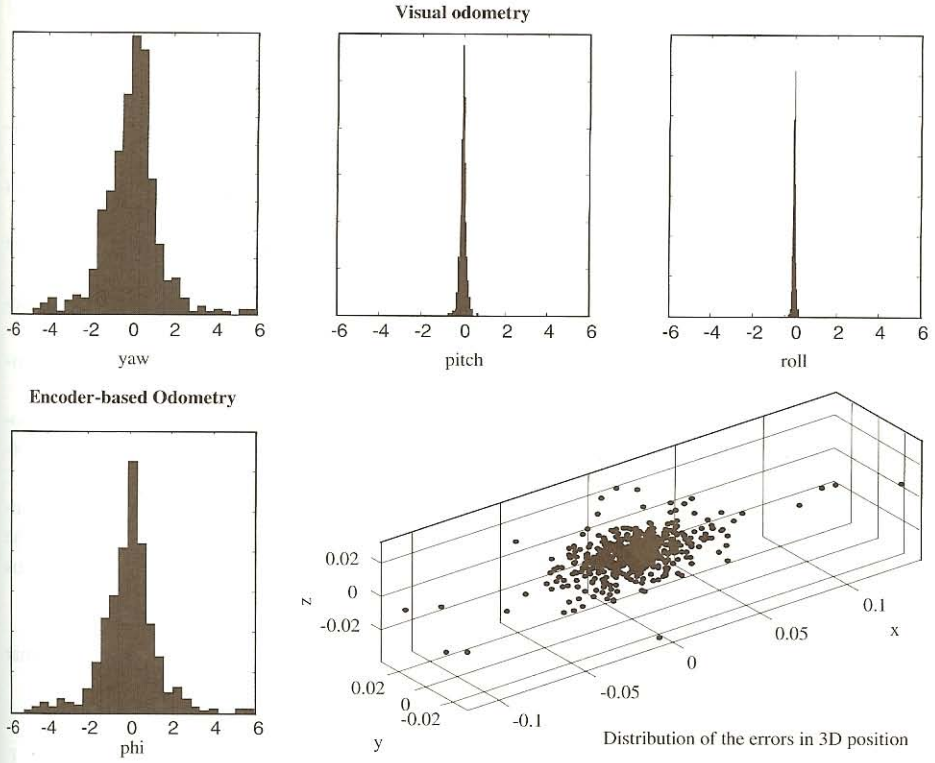

Fig. 4. Histograms of the errors committed in the estimation of the changes in the robot orientation for the visual odometry (top plots) and conventional encoder-based odometry (bottom-left plot) approaches. (bottom-right) Distribution of the errors in the estimation of the change in the robot 3D position for the visual odometry approach.

References

1. Campbell, J., Sukthankar, R., Nourbakhsh, I., Pahwa, A.: A Robust Visual Odometry and Precipice Detection System Using Consumer-grade Monocular Vision. In: A Robust Visual Odometry and Precipice Detection System Using Consumer-grade Monocular Vision, pp. 3421–3427 (2005)
2. Davison, A.J., Reid, I., Molton, N., Stasse, O.: MonoSLAM: Real-Time Single Camera SLAM. IEEE Transactions on Pattern Analysis and Machine Intelligence (2007)
3. Fernandez, D., Price, A.: Visual Odometry for an Outdoor Mobile Robot. In: Conference on Robotics, Automation and Mechatronics, pp. 816–821 (2004)
4. Hahnel, D., Burgard, W., Fox, D., Thrun, S.: An efficient fastslam algorithm for generating maps of large-scale cyclic environments from raw laser range measurements. In: Proc. of Int. Conference on Intelligent Robots and Systems (IROS) (2003)
5. Harris, C.J., Stephens, M.: A combined edge and corner detector. In: Proceedings of 4th Alvey Vision Conference, Manchester, pp. 147–151 (1988)

6. Horn, B.K.P.: Closed-form solution of absolute orientation using unit quaternions. Journal of the Optical Society of America A 4, 629–642 (1987)
7. Kitchen, L., Rosenfeld, A.: Gray-level corner detection. Pattern Recognition Letters 1, 95–102 (1982)
8. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
9. Matthies, L., Shafer, S.A.: Error modeling in Stereo Navigation. IEEE Journal of Robotics and Automation RA-3(3) (1987)
10. Nistér, D., Naroditsky, O., Bergen, J.: Visual Odometry. In: Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 652–659 (2004)
11. Olson, C.F., Matthies, L.H., Schoppers, M., Maimone, M.W.: Rover navigation using stereo ego-motion. Robotics and Autonomous Systems 43(4), 215–229 (2003)
12. Shi, J., Tomasi, C.: Good features to track. Proc. Computer Vision and Pattern Recognition, 593–600 (1994)
13. Sim, R., Elinas, P., Griffin, M., Little, J.J.: Vision-based SLAM using the Rao-Blackwellised Particle Filter. In: IJCAI Workshop Reasoning with Uncertainty in Robotics, Edinburgh, Scotland (2005)
14. Stachniss, C., Grisetti, G., Burgard, W.: Recovering Particle Diversity in a Rao-Blackwellized Particle Filter for SLAM After Actively Closing Loops. In: Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA), IEEE Computer Society Press, Los Alamitos (2005)
15. Thrun, S., Burgard, W., Fox, D.: Probabilistic Robotics. MIT Press, Cambridge (2006)
16. Wang, H., Yuan, K., Zou, W., Zhou, Q.: Visual Odometry Based on Locally Planar Ground Assumption. In: Int. Conference on Information Acquisition, pp. 59–64 (2005)