



UNIVERSITEIT
VAN
AMSTERDAM

Submitted to: internal

IAS technical report IAS-UVA-09-05

The Planar two point algorithm

Olaf Booij and Zoran Zivkovic

Intelligent Systems Laboratory Amsterdam,
University of Amsterdam
The Netherlands

Vision-based localization, mapping and navigation is often performed by searching for corresponding image points and estimating the epipolar geometry. It is known that the possible relative poses of a camera mounted on a mobile robot that moves over a planar ground floor, has two degrees of freedom. This report provides insight in the problem of estimating the exact planar robot pose difference using only two image point correspondences. We describe an algorithm which uses this minimal set of correspondences termed the Two-point algorithm. It is shown that sometimes two non-degenerate correspondences do not define a unique relative robot pose, but lead to two possible real solutions. The algorithm is especially useful as hypothesis generator for the well known RANSAC (RANDOM SAMPLE CONSENSUS) method. The algorithm is evaluated using both simulated data and data acquired by a mobile robot equipped with an omnidirectional camera. The improvement over existing methods is analogous to the improvement of the well known Five-point algorithm over other algorithms for general non-planar camera motions.

Keywords: computer vision, robot localization, robot pose estimation

IAS

intelligent autonomous systems

Contents

1	Introduction	1
2	Related work	1
3	Problem description	2
3.1	Two degrees of freedom, two point correspondences	2
3.2	Two is better than three	2
4	Two-Point Algorithm	3
4.1	Implementation	6
5	Experiments	7
5.1	Simulated data	7
5.1.1	Robustness against mismatches	7
5.1.2	Robustness against non planar motion	8
5.1.3	Partial solutions	8
5.2	Real home environment	9
6	Conclusion	10

Intelligent Autonomous Systems

Informatics Institute, Faculty of Science
University of Amsterdam
Kruislaan 403, 1098 SJ Amsterdam
The Netherlands

Tel (fax): +31 20 525 7461 (7490)

<http://www.science.uva.nl/research/isla/>

Corresponding author:

Olaf Booij
tel: +31 20 525 7536
O.Booij@uva.nl
obooy@science.uva.nl

1 Introduction

Robots are often equipped with a vision sensor to perform a wide variety of tasks, including basic robot tasks such as localization [11], navigation[3] and mapping[6]. At the basis of these tasks is the ability of the robot to estimate its relative pose. A common approach to the problem of vision based pose estimation is by sets of image point correspondences. From these point correspondences the epipolar geometry can be determined, which can be decomposed into the 3d direction of the translation and 3d rotation of the robot.

In the domain of mobile robotics the motion is usually more constrained, because the robots often drive over a more or less planar ground floor. It is known that in this situation only two point correspondences are needed to estimate the relative robot pose [15]. In [9] the existence of a closed form algorithm that uses the minimal amount of two correspondences was first proposed and briefly described. To our knowledge this is the only algorithm known to solve the planar two point problem.

In this report we give the full derivation of this closed form algorithm. We make the novel insight that the number of possible solutions for the problem is sometimes two and thus the algorithm sometimes results in two possible sets of parameters. Finally, we compare the algorithm with a variation of the well known 8-point, which uses 3 point correspondences and can be considered as the standard algorithm to estimate the planar from two images. [5, 11, 15]

For evaluation we use simulated data as well as data acquired by a mobile robot taken in a real home environment. For all datasets we use a omnidirectional vision sensor constructed from an ordinary camera and a hyperbolic mirror [2]. Nevertheless, the algorithm could just as well be applied to other vision systems including conventional cameras.

The rest of the article is structured as follows. First in Section 2 we give an overview of related work. Then in Section 3 it is explained why two point correspondences are sufficient for planar pose estimation and why an algorithm using only two correspondences is useful. In Section 4 the complete derivation of the algorithm is given. The experiments for evaluating the algorithm are described in Section 5. Finally we discuss the results of the experiments and draw conclusions in Section 6.

2 Related work

The common approach of determining the relative pose using two images taken by a calibrated camera is by estimating the Essential matrix [12]. Brooks describes how the Essential matrix can be simplified in case of different motion constraints of stereo camera rig [5]. One such constraint is where the camera rotation axis is perpendicular to the baseline of the rig, which corresponds to the planar motion constraints. This simplified Essential matrix can be estimated using 3 point correspondences and is used in various robot localization and navigation systems [15, 11, 3].

In [15] Ortín made the observation that for determining the planar motion of a calibrated camera only 2 point correspondences are needed. Among the various proposed methods for motion estimation one algorithm uses this insight. However, it uses an iterative method to find a solution and a proof of convergence is omitted.

Goedemé does propose a closed form solution for the planar estimation problem using two correspondences [9]. However, the method is only briefly described and some of the insights necessary for implementing and using the method are omitted. In this report we describe and evaluate the method thoroughly.

A group of algorithms that is strongly related to robot pose estimation tries to compute a so called homing vector given an image of the goal position [1, 8]. These heuristic methods only use the horizontal projection of the landmarks and therefore cannot compute the exact relative robot poses given any number of correspondences (see Section 3) [8]. This is also due to the

use of cone or sphere shaped mirrors which create a panoramic view but make reprojection of images points impossible[2]. In our experiments we use a hyperbolic mirror which does preserve centrality of the vision system.

3 Problem description

This section describes the problem of estimating the relative planar robot pose using two images and explains why it is useful to have an algorithm that uses only two correspondences.

3.1 Two degrees of freedom, two point correspondences

For general motion the problem of estimating the relative pose given two images has 5 degrees of freedom, namely 3 for a 3d translation, plus 3 for a 3d rotation, minus 1 for the scale ambiguity. Every point correspondence in the images can remove one degree of freedom by providing one variable of information, namely 4 for the 2D pixel-locations for both images, minus 3 for the unknown 3d position of the corresponding world point). So for general motion, in order to solve the relative pose problem at least 5 point correspondence are needed. Hence the wide use of the Five-Point Algorithm proposed by Nister [14]. If the 5 correspondences lie in a degenerate formation, for example if 4 of them were caused by landmarks lying on a plane, then an infinite amount of solutions is possible. We will ignore these cases for the rest of the report. The 5 points do not define a unique solution but can be the result of up to 10 possible relative poses. As we will see in Subsection 3.2 this does not make the algorithm useless.

When we restrict the camera to move over a planar surface the number of degrees of freedom of the motion drops to 2, namely 2 for a 2d translation, 1 for a 2d rotation, minus 1 for the scale ambiguity. Using the same reasoning as for the general motion, this problem with 2 degrees of freedom can be solved using the minimum of 2 point correspondences. As we shall see in Section 4 the number of solutions for the relative pose problem will be 2.

Note that when we only take the horizontal image point coordinates into account (as is the case in homing algorithms) then we can not extract any quantitative information from seen landmarks. Every point correspondence would give 2 pieces of information minus 2 for the unknown position planar position of the landmark.

3.2 Two is better than three

It is evident that if only two point correspondences are available, then an algorithm using only these two correspondences is necessary to compute the relative camera pose. However, salient point matching techniques, such as the SIFT (Scale Invariant Feature Transform) method, usually generate in the order of a thousand point correspondences[13]. Depending on the difference between the two images with respect to light conditions, view angle and changes in the environment, the set of point correspondences will have a certain percentage of mismatches. This can be resolved by using a robust estimator that can cope with a high percentage of outliers. A method that is particularly robust against a high percentage of outliers is the RANSAC algorithm (RANdom SAMple Consensus) [16, 10], which nowadays is the standard method for estimating the epipolar geometry from images.

RANSAC estimates a large number of candidate solutions for the epipolar geometry problem and chooses that solution that agrees with the highest number of point correspondences. Each candidate is computed given a randomly selected subset of point correspondences. Each candidate is then used to compute an error for the complete set of correspondences. The candidate with the highest number of correspondences with an error below a hard set threshold, is picked as the best fitting model. A common choice for the error measure is the residual error which is

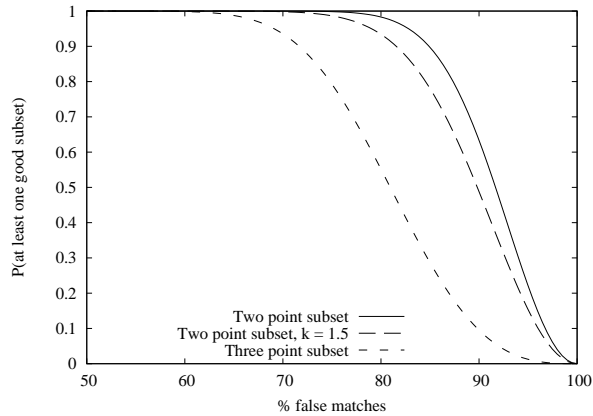


Figure 1: Comparison of the performance of the RANSAC when using a two point algorithm and a three point algorithm. The number of candidates N was set to the typical value of 100. Two versions of two points are shown, one for an algorithm which provides a unique solution and one which provides 2 solutions in 50% of the time ($k = 1.5$), which is more realistic as explained in Section 4.

easy to compute using the epipolar geometry or the somewhat more involved Sampson distance (see [16, 10] for a discussion of these measures).

The rationale of RANSAC is that at least one of the candidate solutions is computed from only good point correspondences and will thus have a large number of fitting correspondences in the complete set. The chance of RANSAC picking at least one subset of only good feature matches in a certain amount of iterations N is given by:

$$p = 1 - (1 - w^s)^{\binom{N}{k}}, \quad (1)$$

where w the percentage of good correspondences, s the size of the subset and k the number of solutions for one iteration. As can be seen p depends exponentially on s , so it is beneficial to choose it as small as possible. And that is why it is important to use an algorithm that needs the minimal number of correspondences. In Figure 1 Equation 1 is used to show the advantage of using an algorithm that uses two correspondences as opposed to the standard algorithm that uses three correspondences. As can be seen the largest gain of using a two point algorithm is for a high percentage of outliers between 70% and 95%

4 Two-Point Algorithm

This section gives the complete derivation of the Two Point Algorithm. We assume that two images were taken at two different location L and R and that in both images we detect two landmarks named F_1 and F_2 , see Figure 2. The goal is to compute the relative heading θ and the relative rotation ω as shown in Figure 2(a).

For both landmarks we can compute the vertical and horizontal angles of the light rays as they hit the camera surface using the camera model of the used vision system. We denoted the vertical angles with an α , α_{L1} and α_{L2} for the landmarks seen in camera L and α_{R1} and α_{R2} for the landmarks seen in camera R (see Figure 2(c)). The horizontal angles are denoted with β 's using the same scheme (see Figure 2(a)). Note that we placed a minus in front of β_{R2} to indicate that the angle from R to F_1 is negative in this example.

We will now try to find enough parameters of the quadrangle (also known as a tetragon)

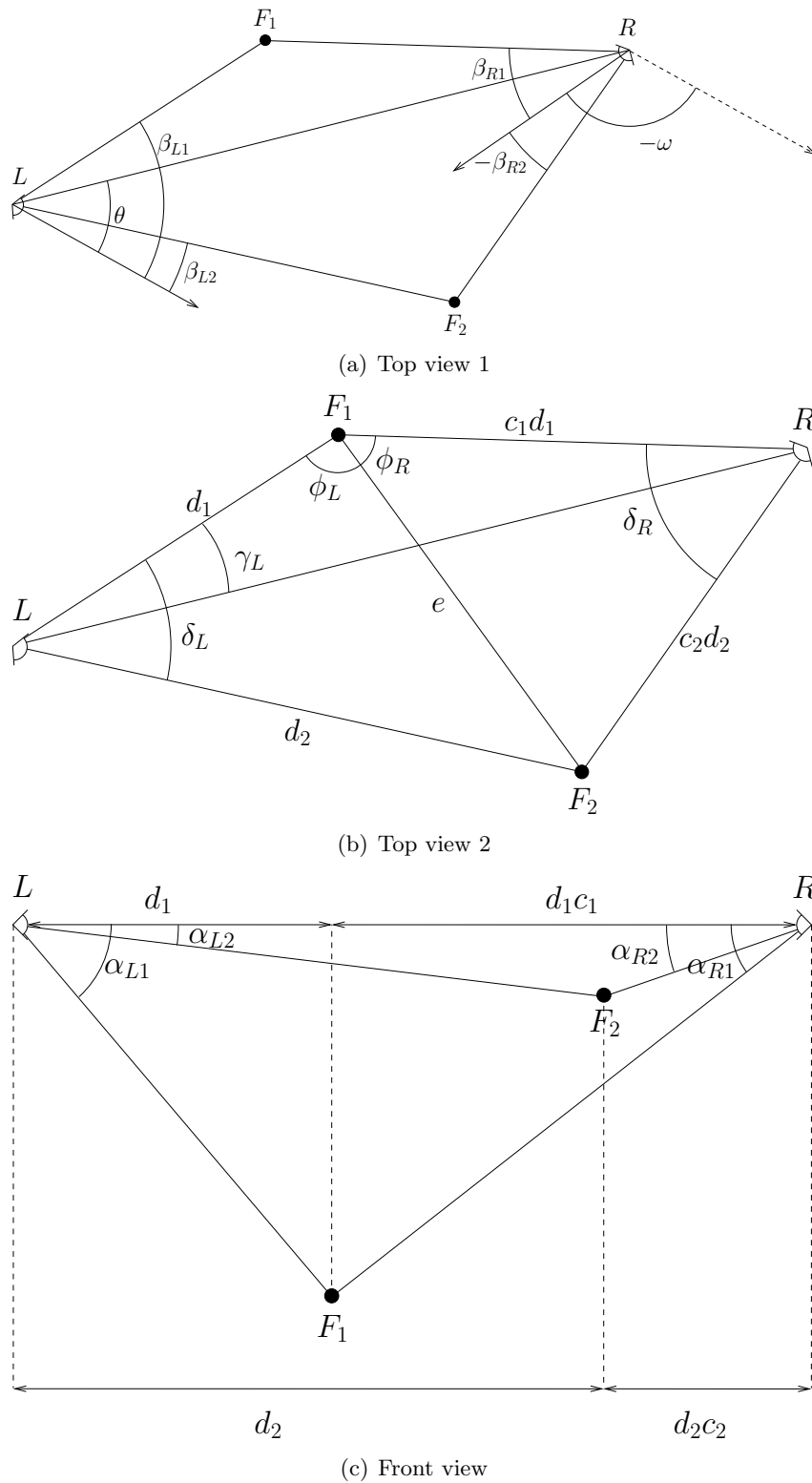


Figure 2: Schematic drawings showing the variables involved in the two-point algorithm. (a),(b) both top-views in which distances and angles are measured only in 2d (parallel to the ground-floor). (c) A frontal view, in which the α 's are the downward looking angles and horizontal lines denote the distances in 2d (parallel to the ground-floor).

formed by the points L - F_2 - R - F_1 , as seen in Figure 2(b), to compute the angles of the triangle between one of the landmarks and the robot poses. In the example we chose the triangle L - F_1 - R .

We can compute the angles of the quadrangle at L and R by taking the differences δ_L and δ_R between both pairs of β 's for the cameras:

$$\delta_L = \beta_{L2} - \beta_{L1} \quad (2)$$

$$\delta_R = \beta_{R2} - \beta_{R1} \quad (3)$$

The distances from the two views to the landmarks can not be determined, but the ratio between these distances can be determined using the vertical view angles. In Figure 2(c) it can be easily seen that the ratio between the distance from view L to landmark F_1 and from view R to landmark F_1 , denoted by c_1 , is the same as the ratios of the tangents of the vertical view angles. The same holds for landmark F_2 :

$$c_1 = \frac{\tan(\alpha_{L1})}{\tan(\alpha_{R1})} \quad (4)$$

$$c_2 = \frac{\tan(\alpha_{L2})}{\tan(\alpha_{R2})} \quad (5)$$

We now have enough parameters to determine the shape of the quadrangle, except for the scale which can not be determined from point correspondences. Without loss of generality we set one of the distances, namely the one from view L to landmark F_1 , to 1

$$d_1 \equiv 1 \quad (6)$$

The ratio between distances from view L to landmarks F_1 and F_2 can be computed with the use of the distance between landmark F_1 and F_2 denoted with e . The distance e can be expressed in two ways using the left triangle L - F_1 - F_2 and the right triangle R - F_1 - F_2 by applying the Law of Cosines:

$$e^2 = 1 + d_2^2 - 2d_2 \cos(\delta_L) \quad (7)$$

$$e^2 = c_1^2 + c_2^2 d_2^2 - 2c_1 c_2 d_2 \cos(\delta_R) \quad (8)$$

Equating the right parts of equations 7 and 8 gives a second order polynomial in d_2 :

$$1 + d_2^2 - 2d_2 \cos(\delta_L) = c_1^2 + c_2^2 d_2^2 - 2c_1 c_2 d_2 \cos(\delta_R) \quad (9)$$

$$(1 - c_2^2) d_2^2 - 2(\cos(\delta_L) - c_1 c_2 \cos(\delta_R)) d_2 + 1 - c_1^2 = 0 \quad (10)$$

This can be solved using the quadratic formula:

$$a = 1 - c_2^2 \quad (11)$$

$$b = -2(\cos(\delta_L) - c_1 c_2 \cos(\delta_R)) \quad (12)$$

$$c = 1 - c_1^2 \quad (13)$$

$$d_2 = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (14)$$

As can be seen this results in two solutions for d_2 , which indicates that given two point correspondences there are two possible solutions for the relative pose. The rest of the derivation has to be performed using both these values.

Using the distances d_1 and d_2 we can calculate distance between the two landmarks:

$$e = \sqrt{1 + d_2^2 - 2d_2 \cos(\delta_L)} \quad (15)$$

For one of the two possible values for d_2 , e can turn out to be imaginary. In that case only one solution can be computed. This happens in exactly 50% of percent of the cases. A proof for this is omitted.

Using again the Law of Cosines the angles ϕ_L and ϕ_R can be calculated:

$$\phi_L = \arccos\left(\frac{d_2^2 - e^2 - 1}{-2e}\right) \quad (16)$$

$$\phi_R = \arccos\left(\frac{c_2^2 d_2^2 - e^2 - c_1^2}{-2c_1 e}\right) \quad (17)$$

The angle $L-F_1-R$ is not always a simple addition of these two angles as is the case in the given example. It depends on the shape of the quadrangle:

$$\phi = \phi_L + \phi_R \quad \text{if } \delta_L < \pi \text{ and } \delta_R > \pi \quad (18)$$

$$\phi = \phi_L - \phi_R \quad \text{if } \delta_L < \pi \text{ and } \delta_R < \pi \quad (19)$$

$$\phi = -\phi_L + \phi_R \quad \text{if } \delta_L > \pi \text{ and } \delta_R > \pi \quad (20)$$

$$\phi = -\phi_L - \phi_R \quad \text{if } \delta_L > \pi \text{ and } \delta_R < \pi \quad (21)$$

From the angle ϕ we can compute γ_L using the atan2 function to get an angle in the range $[0, 2\pi)$:

$$\gamma_L = \text{atan}(c_1 \sin(\phi), 1 - c_1 \cos(\phi)) \quad (22)$$

Using the horizontal angles of the landmarks as seen in the images we can now compute the heading:

$$\theta = \beta_{L1} + \gamma_L \quad (23)$$

And the rotation is computed using the quadrangle with the points L , $F1$, R and the intersection of the direction of R and L :

$$\omega = -\beta_{R1} + \beta_{L1} - \phi \quad (24)$$

4.1 Implementation

The complete algorithm as given by equations (2)-(5) and (11)-(24), taking account of the two possible solutions, can be applied on every two distinct point correspondences. However for a robust implementation which handles noisy point correspondences and mismatches one has to be cautious. Most implementations of the arc cosine function return a complex number if called with a variable which has an absolute value higher than 1. This could happen for some noisy pair of point correspondences.

Also it could happen that c_1 or c_2 become negative if a point correspondence resulting from a mismatch has a positive vertical angle in one view and a negative vertical angle in the other. These mismatches can be discarded by checking the sign of c_1 and c_2 .

When the algorithm is applied in RANSAC, the residual or the Sampson distance can be used to compute the error for a set of point correspondence given a heading and a rotation. Usually this is done through a parameterization of the epipolar geometry called the essential matrix. This 3x3 matrix can be constructed as follows:

$$E = \begin{bmatrix} 0 & 0 & \sin(\omega - \theta) \\ 0 & 0 & \cos(\omega - \theta) \\ \sin(\theta) & -\cos(\theta) & 0 \end{bmatrix} \quad (25)$$

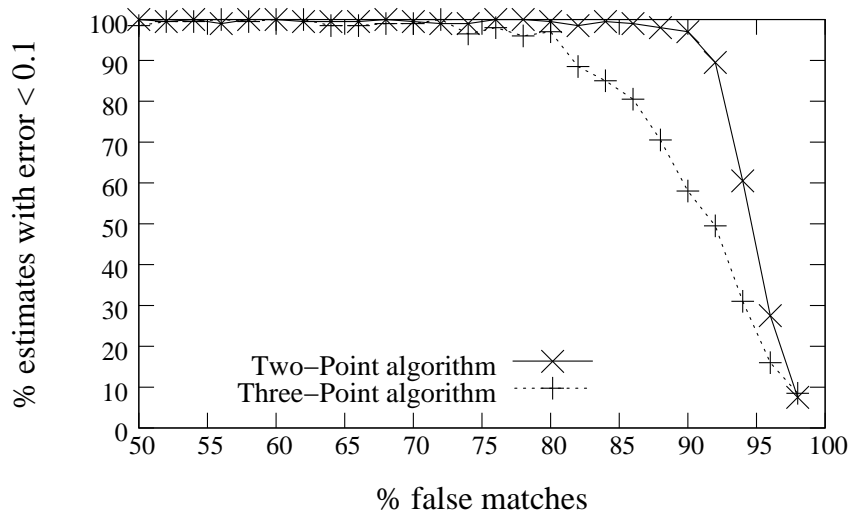


Figure 3: Comparison of robustness of the Two-Point algorithm and the Three-Point Algorithm to a increasing percentage of outliers.

5 Experiments

The Two-Point algorithm is evaluated using both simulated data and data acquired by a our robot in a challenging home environment. It is compared to the commonly used variant of the well-known 8 point algorithm, which also assumes planar motion (see [15]).

For both methods the estimated heading of the translation is used for comparison, while it is much more sensitive to noise than the rotation [14]. The error of the estimated heading is computed by taking the absolute difference with the ground truth heading, which is available in both sets. Because these error values are not normally distributed, the standard deviation is not very descriptive. Instead we use a different error measure, namely the percentage of estimates with an error smaller than a threshold.

While omnidirectional vision has proven itself in robot localization tasks and is used more and more, we used a catadioptric vision system for capturing images. For the simulated data we simulate our omnidirectional vision system. The upward view angle is 28 degrees and the downward view angle 55 degrees. Horizontally the view is 360 degrees

5.1 Simulated data

The simulator consists of a uniformly random spherical point cloud of 100 3D points. Two view points are randomly picked from a circle that lies in the center of the point cloud with a radius two times smaller than that of the point cloud.

All the points in the point cloud are projected on a cylinder shaped panorama with a radius of 1 around the viewing point where a certain amount of noise added. A set of point correspondences is then constructed by matching a certain amount of image points that are projections of different points in the point cloud, thus artificially adding mismatches to the set.

5.1.1 Robustness against mismatches

We now vary the number mismatches that are added to the set while keeping the noise on a fixed value of 0.2. In Figure 3 we see that the Two-Point algorithm has the same good performance as the Three-Point algorithm until the percentage of mismatches crosses the 80%. At 90% also the performance of Two-Point algorithm starts to drop.

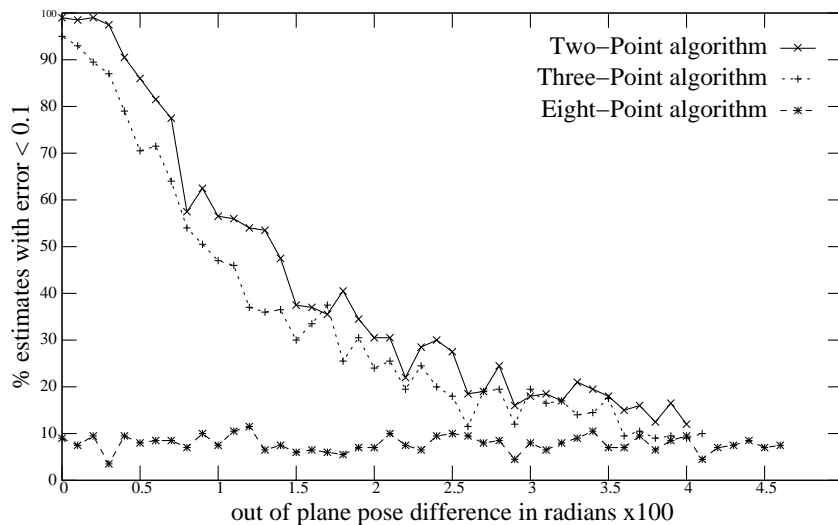


Figure 4: Comparison of robustness of the Two-Point algorithm and the Three-Point Algorithm to a increasing non planar motion.

The graph looks very similar to the theoretical result depicted in Figure 1. While the chance of picking 2 good correspondences with 90% outliers is still at 97%, the change of picking 3 good correspondences has dropped to 58%.

5.1.2 Robustness against non planar motion

Another interesting experiment is to see what happens if the assumption of planar motion is not valid. In real applications there will never be a completely flat surface. So we add an varying amount of non planar pose difference to the robot poses. The noise level was again set to 0.2 and the percentage of mismatches to 80%. In Figure 4 we see that the performance of both the Two-Point algorithm and the Three-Point algorithm decrease fast under non planar motion. With 1 degree of non planar motion (0.017radians) the number of good estimations has halved. An angular change of one degree seems small, but for roads this means a change of 2% in steepness, which is quite uncommon. Also the difference between the Two-Point Algorithm and the Three-Point Algorithm decreases with more out of plane motion.

As a reference, we have also plotted the performance of the well known 8 point algorithm for general motion [16, 10], which does not assume planar motion, hence it is not influenced by the non planar motion. Nevertheless the performance of the 8 point algorithm is worse because it has more parameters to estimate and can not cope with the high number of mismatches.

As can be seen both planar pose estimators decrease to the same level as the more general estimator after adding 3 degrees of out of plane motion.

5.1.3 Partial solutions

Because the solution space is only 2 dimensional, we can depict it easily, see Figure 5, with the axes corresponding to the heading θ and the rotations ω . As explained in Section 3 every point correspondence removes one degree of freedom. In the figure it becomes clear what this entails. We simulated two robot poses with the same rotation and heading zero. For various random world points and its resulting point correspondence we plotted the curve of all possible robot poses that could have caused that correspondence. The curves were generated by sampling a second mismatching correspondence randomly and feeding them to the Two-point algorithm.

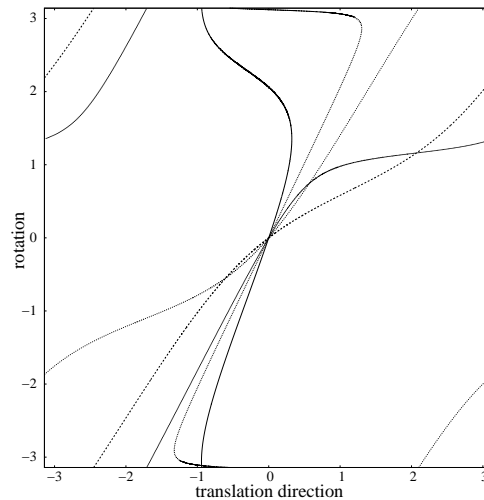


Figure 5: Depicted is the complete solution space with all possible headings θ on the horizontal axis and rotations ω on the vertical axis. Every curve through solution space resulted from one point correspondence.

The curves all intersect in the correct solution for the robot pose with zero heading and zero rotation. When looking closely it can be seen that some pairs of curves intersect each other a second time somewhere in solution space. This again shows that two correspondences define two possible solutions.

5.2 Real home environment

The algorithm was also tested on a dataset taken in a real home environment. The omnidirectional images were acquired under changing light conditions while the robot was driving around. Because the robot was heavily loaded it leaned over during driving. In total 401 images were taken¹ Besides images the robot also took laser range scans which were used as input for a SLAM algorithm providing us with accurate ground truth pose information[7]. These were used to compute the ground truth heading for evaluation.

From the images random pairs were picked for evaluation. Point correspondences between the two images were found as follows. First Harris corners were extracted from both images using the Intel OpenCV library and a 81-dimensional descriptor of the corner point was formed by taking an image patch of 9 by 9 pixels around it. Then matching features were found between the images by finding the first and second nearest neighbor in feature space using the Euclidean distance. Two features match if the first nearest neighbor is at least 0.8 closer than the second nearest neighbor, as described in [13].

Again the Two-Point algorithm and the Three-Point algorithm are applied, see Figure 6 for the resulting inlying features of an example image pair. Results from image pairs acquired with a distance of more than 5 meters were discarded, as the visual overlap was too small. For image pairs taken at a distance smaller than 5 meters the average(std deviation) percent of pose estimates with an error below .5 is 53.5(21.7) for the Three-Point algorithm and 55.1(21.1) for the Two-Point algorithm.

The performance of Two-point algorithm was slightly better than that of the Three-point algorithm. However, the difference was not significant. The performance increase as apparent for the simulated data could not be reached for the real robot data. This is possibly due to the

¹This dataset is available at <http://www2.science.uva.nl/sites/cogniron/fs2hsc/Data/Home1/run1/>.

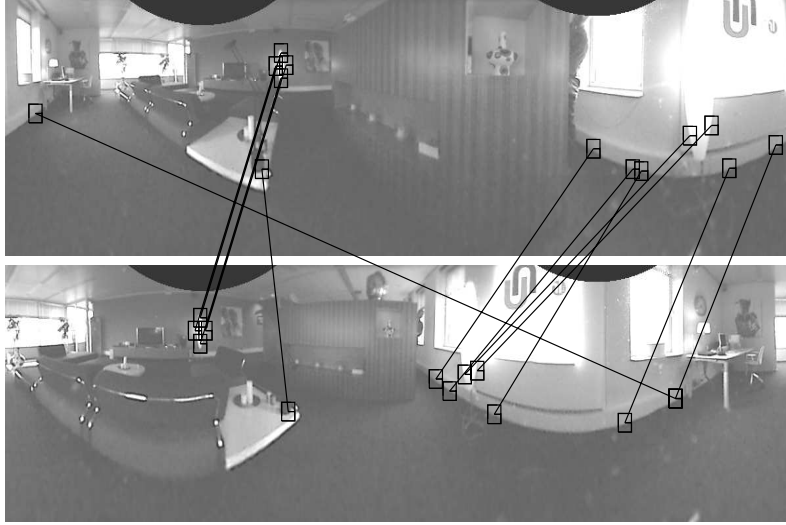


Figure 6: An example pair of images. The boxes indicate the Harris features and the lines denote features that satisfy the relative orientation found by the Two-Point algorithm.

fact that the robot leaned over while driving through the environment violating the planarity assumption. Another possible cause is the relatively high number of inliers. Although the lighting conditions were challenging the environment was static and feature rich.

6 Conclusion

We investigated the problem of estimating the relative planar robot pose given two images. An algorithm was given that computes this relative pose using only 2 point correspondences found in the images. We have shown that the algorithm is robust to a high number of mismatching point correspondences if used in the hypothesize and verify scheme of RANSAC. It is much more robust than the now standard algorithm used for relative pose estimation under the planar constraint, the Three-Point algorithm.

It is shown that the problem does sometimes not define a single robot pose, put 2 real robot poses. This should be resolved using a robust estimator, for example RANSAC. Another interesting idea is to use a Hough transform to find a robust solution. Because the search space has only 2 dimensions, maintaining a histogram would not be very demanding, as would be the case for general non planar motion. This idea has been recently developed in [4].

Further experiments show that the performance of both the standard algorithm and the described algorithm decreased to that of the general non planar 8-point algorithm if planarity assumption is violated with more than 5 degrees of rotation. This is probably also the reason why it is not significantly better than the Three-Point algorithm for the conducted real robot experiments. If performance of the algorithm suffers from the load of the robot, then it is better to stop driving while taking images.

References

- [1] A. A. Argyros, K. E. Bekris, S. C. Orfanoudakis, and L. E. Kavraki. Robot homing by exploiting panoramic vision. *Autonomous Robots*, 19(1):7–25, 2005.

-
- [2] Simon Baker and Shree K. Nayar. A theory of single-viewpoint catadioptric image formation. *Int. J. Comput. Vision*, 35(2):175–196, 1999.
- [3] O. Booiij, B. Terwijn, Z. Zivkovic, and B. Kröse. Navigation using an appearance based topological map. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, pages 3927–3932, Roma, Italy, April 2007. IEEE.
- [4] O. Booiij, Z. Zivkovic, and B. Kröse. A map-estimator for vision based planar pose estimation. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, Anchorage, Alaska, May 2010. IEEE. (Submitted).
- [5] M. Brooks, L. de Agapito, D. Huynh, and L. Baumela. Towards robust metric reconstruction via a dynamic uncalibrated stereo head. *Image Vision Comput.*, 16(14):989–1002, 1998.
- [6] R. M. Eustice, H. Singh, and J. J. Leonard. Exactly sparse delayed-state filters for view-based SLAM. *IEEE Transactions on Robotics*, 22(6):1100–1114, December 2006.
- [7] J. Folkesson, P. Jensfelt, and H. Christensen. Vision SLAM in the measurement subspace. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, pages 30–35, Barcelona, Spain, April 2005. IEEE.
- [8] M.O. Franz, B. Schlkopf, and H.H. Blthoff. Where did i take that snapshot? scene-based homing by image matching. *Biological Cybernetics*, 79:191–202, 1998.
- [9] T. Goedemé, T. Tuytelaars, G. Vanacker, M. Nuttin, and L. Van Gool. Feature based omnidirectional sparse visual path following. In *Intl. Conf. on Intelligent Robotics and Systems*, pages 1003–1008, Edmonton, Canada, August 2005. IEEE/RSJ.
- [10] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision, second edition*. Cambridge University Press, 2003.
- [11] J. Kosecká, F. Li, and X. Yang. Global localization and relative positioning based on scale-invariant keypoints. *Robotics and Autonomous Systems*, 52(1):27–38, 2005.
- [12] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, Sept. 1981.
- [13] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. Journal of Computer Vision*, 60(2):91–110, 2004.
- [14] David Nistér. An efficient solution to the five-point relative pose problem. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(6):756–777, 2004.
- [15] D. Ortín and J. M. M. Montiel. Indoor robot motion based on monocular images. *Robotica*, 19(3):331–342, 2001.
- [16] P. H. S. Torr and D. W. Murray. The development and comparison of robust methods for estimating the fundamental matrix. *Int. Journal of Computer Vision*, 24(3):271–300, 1997.



IAS reports

This report is in the series of IAS technical reports. The series editor is Bas Terwijn (B.Terwijn@uva.nl). Within this series the following titles appeared:

C. Dimitrakakis *Bayesian variable order Markov models: Towards Bayesian predictive state representations* Technical Report IAS-UVA-09-04, Informatics Institute, University of Amsterdam, The Netherlands, September 2009.

C. Dimitrakakis and C. Mitrokotsa *Statistical decision making for authentication and intrusion detection* Technical Report IAS-UVA-09-03, Informatics Institute, University of Amsterdam, The Netherlands, August 2009.

P. de Oude and G. Pavlin *Dependence discovery in modular Bayesian networks* Technical Report IAS-UVA-09-02, Informatics Institute, University of Amsterdam, The Netherlands, June 2009.

C. Dimitrakakis *Complexity of stochastic branch and bound for belief tree search in Bayesian reinforcement learning* Technical Report IAS-UVA-09-01, Informatics Institute, University of Amsterdam, The Netherlands, April 2009.

All IAS technical reports are available for download at the ISLA website, <http://www.science.uva.nl/research/isla/MetisReports.php>.