

Underwater Robot Localization using Artificial Visual Landmarks¹

Pifu Zhang^{†2} Evangelos E. Milios[†] Jason Gu[‡]

[†]Faculty of Computer Science

[‡]Department of Electrical and Computer Engineering

Dalhousie University

Halifax, Canada, B3H 1W5

{pifu,eem}@cs.dal.ca jason.gu@dal.ca

abstract

Accurate localization for underwater robot is a big challenge in the mobile robot community. We will use acoustic and visual-based approaches to solve the problem. In this paper, we focus on the artificial landmark visual-based localization. The robot carries the camera to localize its position by calculating the camera's viewpoint through looking at the landmarks. The position of landmarks is known in a world-centered coordinate system (WCCS). A method is presented in this paper to recover the camera's viewpoint with minimum three feature points and a single image from one camera. Two steps are used in this paper: first step is to calculate the feature points' 3D coordinates in a camera centered coordinate system (CCCS); second step is to obtain a closed-form solution through the geometric transformations to map the 3D points from CCCS to WCCS. The algorithm is robust and efficient. It uses the fewest feature points required so far to deal with the same problem.

Keywords: localization, landmark, robot navigation

1 Introduction

The Autonomous Underwater Robot, also called Autonomous Underwater Vehicle (AUV), usually

works in the unknown environment for sea bed survey, salvage, ocean species monitoring, and disaster operation. In order to ensure it works properly, it is important to know its position with respect to a map or landmarks at known positions.

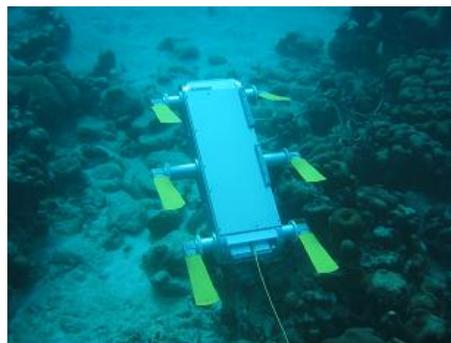


Figure 1 The robot underwater off the Barbados west coast

There are two main kinds of approaches for the localization of AUV: acoustic and visual method. The acoustic method [7] uses hydrophone, long baseline (LBL), short baseline (SBL) or ultra short baseline (USBL) system to measure the time flying of sound which generated by the AUV, and then to determine the position of the AUV. The visual method can be classified into two types: there are methods that explicitly recover 3D information using stereo and others that use a single camera.

In our robot (Fig. 1), we will use both of the acoustic and visual methods for its localization. On

¹Funding for this work was provided by NSERC Canada and IRIS NCE

²The author is also with Computer Science Department, Changsha University of Science and Technology

the working area, a set of buoys (Fig. 2) will be set on the surface of the water. On each of the buoys, GPS and compass will be mounted to measure its position and orientation; and four hydrophones and four landmarks (LED) will be mounted, which have the coordinates in a world-centered coordinate system (WCCS) $P_i = (x_i, y_i, z_i)^T, (i = 1, \dots, n)$. Localization estimates from both the visual and the acoustic system will be fused by the consistent pose estimation (CPE) system.



Figure 2 The buoy for the robot localization

In this paper, we introduce the method for the localization of underwater robot using landmarks with a single image. In robot working area, we can not put too much buoys in the ocean environment. The reasons are that too many buoys will affect the proper working of the robot, and are expensive. So we will use two buoys for our system. In most of cases, due to the limitation of the view angle, the camera can take only one buoy's image. So we must find a solution for the 3D localization with the smallest number of visual landmarks.

The problem can be stated formally as follows: *Given a set of m control points whose 3-dimensional coordinates are known in a world coordinate system, and given an image (taken by a calibrated camera) in which some subset of the m control points are visible, determine the location from which the image was obtained.*

Many researchers have extensively investigated the problem in the previous two decades. Most of the approaches attempting to solve the problem can be classified into two groups: (1) closed form solutions [3, 1], and (2) numerical solutions [2, 8, 6].

We use two steps to solve the problem: The feature points in camera centered coordinate system (CCCS) can be derived from the camera model and

their geometric constraints; and these 3D points based on CCCS can be transformed to real world coordinate system (WCCS) by a series of geometric transformations. The transformation matrix during this process includes orientation and position of the camera. The algorithm is very simple, robust, and efficient. The most significant contribution in this paper is that we introduce a novel approach, *geometric transformation*, to solve the problem with as few as three feature point correspondences.

This paper is organized as following: in section 2, a critical survey related to the 2D to 3D correspondence is presented; in section 3, we derive the feature point recovery based on CCCS; in section 4, a new approach is introduced to mapping the feature points from CCCS to WCCS; in section 5, experimental results are presented.

2 Literature Survey

The vision based localization of the robot is equivalent to the external camera calibration problem. The solution for the problem can be classified into two approaches. One is using stereo or sequence images to establish constraints, and then get the pose of the camera. Roth [10] and Madjidi [9] proposed a method of computing camera positions from a sequence of overlapping images obtained from a binocular/trinocular camera head.

Se & Lowe [11] solved the local and global localization for the mobile robot by using the SIFT (Scale Invariant Feature Transform) features. Jang [5] proposed a robot self-localization using artificial and natural landmarks by computing the local Zernike moments.

Another approach is using a single image and familiar environment (landmarks) to solve the camera pose problem. Early work was done by Fischler and Bolles [3] who found the closed-form solution by using 3 point correspondences, but there are four solutions. Further information is need to select the right solution. In the book *Robot Vision*, Horn describes an approach based on a set of non-linear equations among the camera viewpoint and feature points based on the basic camera model and perspective projection [4]. Correspondences between three points pairs between image and world coor-

dinates are the basic requirement to get the pose of camera. Marquardt-Newton iterative method can be used to solve the equations.

Liu et al. [8] examined alternative iterative approaches by using line and point correspondences. They can solve the problem with only three line or point correspondences, but the three Euler angles must be less than 30^0 .

In this paper, we will use at least three point correspondences to solve for the camera pose (position and orientation). The method used in this paper is different from the previous method [4, 3, 8]. It is shown to be more stable respect to the initial value than the previous methods, and even efficient and simple.

3 Feature Point Recovery in the CCCS

Assuming there are feature points $P_i = (x_i, y_i, z_i)$, $i = 1, \dots, n$ in the WCCS, which are not collinear. By using the calibrated camera to take an image which includes all the feature points, their corresponding image coordinates can be taken (u_i, v_i) , $i = 1, \dots, n$. From the basic camera model, we have the equation as

$$u_i = \frac{fx'_i}{z'_i} \quad v_i = \frac{fy'_i}{z'_i} \quad (1)$$

here f is the focal length of the camera, $i = 1, \dots, n$, and x'_i , y'_i , and z'_i are the points in the CCCS corresponding to features.

In equation (1), we know that for every feature point correspondence, it is possible to get two constraints, but there are three unknowns. For three feature points, there exist six constraints and nine unknowns. Three additional constraints are needed to solve for the unknowns.

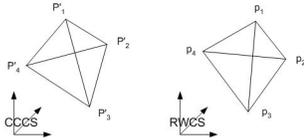


Figure 3 Invariant parameters between the two coordinate systems

By exploiting the relationship between (x'_i, y'_i, z'_i) and (x_i, y_i, z_i) , we note that both of them represent the same feature points in the 3D space, but from different coordinate systems: (x'_i, y'_i, z'_i) is based on

CCCS, while (x_i, y_i, z_i) is based on WCCS. The distance between any two feature points is independent of the coordinate system in which they are described (Fig. 3). Therefore the distance between any two feature points d_{ij} in WCCS is as same as d'_{ij} in CCCS. So if we use the distance between the feature points, we have

$$d_{ij} = d'_{ij}; \quad i, j = 1, \dots, n, \text{ and } i \neq j \quad (2)$$

where

$$d_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \quad (3)$$

So it is possible to solve for the unknowns in the system, which are the coordinates in the CCCS, with only three correspondent points. Since this is a nonlinear system of equations, we will use Newton method to solve it. The initial value can be taken from the landmark's coordinates in WCCS directly.

An alternative approach is to cast the problem as an optimization problem, where the objective function is:

$$F = \sum_{i,j=1,i \neq j}^n (d_{ij} - d'_{ij})^2 \quad (4)$$

If we take

$$\min_{x'_i, y'_i, z'_i} F \quad (5)$$

subject to the constraint equation (1). We can solve this optimization problem with Newton-Lagrange method to the appropriate results.

4 Closed-Form Solution for the Geometric Transformation between CCCS and WCCS

From CCCS to WCCS, there exists an Euclidean transformation. If we use \tilde{P}_i and \tilde{P}'_i to express the homogenous coordinates of P_i and P'_i , the Euclidean transformation can be expressed as

$$\tilde{P}'_i = \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} \tilde{P}_i \quad (6)$$

where R is the 3×3 rotation matrix, and t is the translation.

Our proposed method is based on the geometric invariant of the feature points [12]. It is possible to solve the mapping problem with as few as three points. If there are more than three points, the basic requirement is that all the points are on the approximately planar. If the number of feature points P_i is 3, they define a planar patch; but if the number of feature point P_i is bigger than 3, supposing it is n , it is possible to get a n connected un-overlapped planar patches as following.

Assuming P_c is the center of mass of the set of P_i , we have

$$P_c = \frac{1}{n} \sum_{i=1}^n P_i \quad (7)$$

By using the centroid point P_c and two adjacent point P_i and P_{i+1} , we can define a triangular patch. Over all, we can get n triangle patches which intersect at point P_c . For each of these triangle patches, we can get its normal vector N_i . The average normal vector of all the patches is defined by

$$N = \frac{1}{n} \sum_{i=1}^n N_i \quad (8)$$

We can get the centroid point P'_c and normal vector N' for the set of feature points P'_i in the same manner.

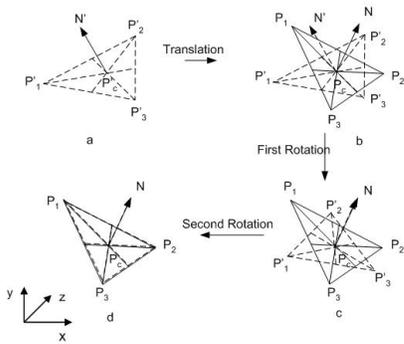


Figure 4 Geometric transformation from CCCS to WCCS in 3 point correspondences case

The set of constructed planar patches for each feature point set P_i and P'_i ($i = 1, \dots, n$) can be expressed by Ω and Ω' in 3D space, respectively. We consider the three point correspondences as an example, in Fig 4. Three steps, one translation and two rotations, will be used to transform the

plane Ω' to plane Ω . First, the plane Ω' is translated from its centroid P'_c to P_c , the centroid of the plane Ω (Fig 4.b). Suppose the translation matrix is M_t , then the new plane can be expressed by $M_t\Omega'$. Second, the plane $M_t\Omega'$ is rotated with P_c as pivot around the axis $N \times N'$ until N' reaches N . Here the rotation angle can be expressed by φ .

After this process, the normal vectors of both patches Ω' and Ω coincide (Fig 4.c). Both of the patches have the same centroid and normal vector, but different orientation on the (common) plane. So it necessary to carry out a second rotation.

The normal vector N of patch Ω is the axis for the second rotation, and the pivot of the rotation is the centroid P_c of patch Ω . The angle for the rotation must be decided. For every feature point correspondences P_i and P'_i , we can get a rotation angle $\angle P_i P_c P'_i = \theta_i$, $i = 1, \dots, n$, so we can get the average rotation angle as

$$\theta = \frac{1}{n} \sum_{i=1}^n \theta_i \quad (9)$$

It must be pointed out that in most of the cases, $\theta_i \neq \theta_j$, where $i = 1, \dots, n$, and $i \neq j$. This is because of the noise of measurement data and optimization approximation in the previous section. So it is necessary to take the average of all the available angles.

Assuming the translation matrix is M_t , the first rotation matrix is expressed by M_{r1} , and the second rotation is expressed by M_{r2} . So the total transformation from plane Ω' to plane Ω is that

$$M = M_{r2} M_{r1} M_t \quad (10)$$

The matrix M is actually the homogenous matrix of external camera calibration. So we got the closed-form solution for rotation matrix $R_{3 \times 3}$ and its correspond translation $t_{3 \times 1}$.

The translation part of the transformation is

$$t_x = M(1,4) \quad t_y = M(2,4) \quad t_z = M(3,4) \quad (11)$$

We know that the Euler angles and the rotation matrix have the following relationship [2]

$$R = \begin{bmatrix} c_2 c_3 & s_1 s_2 c_3 + c_1 s_3 & -c_1 s_2 c_3 + s_1 s_3 \\ -c_2 s_3 & -s_1 s_2 s_3 + c_1 c_3 & c_1 s_2 s_3 + s_1 c_3 \\ s_2 & -s_1 c_2 & c_1 c_2 \end{bmatrix} \quad (12)$$

where $c_1 = \cos\phi_x$, $c_2 = \cos\phi_y$, $c_3 = \cos\phi_z$, $s_1 = \sin\phi_x$, $s_2 = \sin\phi_y$, and $s_3 = \sin\phi_z$. So we can get the Euler angles directly from the following formulas if there is no Gimbal lock state.

$$\phi_y = \arcsin(M(3,1)) \quad (13)$$

$$\phi_z = \arccos(M(1,1)/\cos(\phi_y)) \quad (14)$$

$$\phi_x = \arccos(M(3,3)/\cos(\phi_y)) \quad (15)$$

The algorithm of the camera viewpoint recovery is summarized in Fig. 5.

Input:
(1) camera parameters
(2) coordinate of landmarks
Output: the pose of the camera (robot)
1 compute the feature points based on CCCS
2 mapping between CCCS and WCCS
2.1 translation from P'_c to P_c
2.2 rotation Ω' around $N \times N$ with angle φ
2.3 rotation with new Ω' around N with angle θ
2.4 get the transformation matrix M
3 calculate results
3.1 take the location from M directly (eq. (11))
3.2 calculate orientation from M by (eq. (13-15))

Figure 5 Procedure for the algorithm

5 Experimental Results

5.1 Experiment in Lab

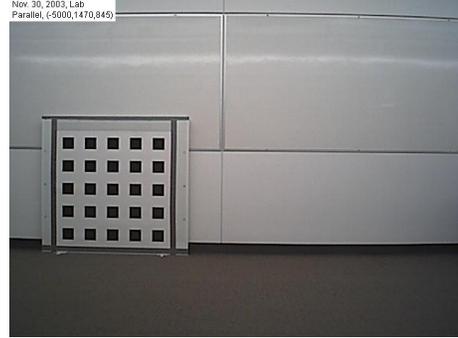


Figure 6 Image which was taken from lab

We calibrated the camera in the air using a standard calibration pattern. We then used the calibration pattern as our artificial landmark set, and we solved the camera localization problem for several camera positions (Fig. 6).

By using our algorithm, we take only 3 outmost corners as landmarks. Table 1 shows the recovered camera poses in comparison with the actual pose. In all eight cases of the experiment, the position estimation error, measured by the distance between the actual and estimated point, is less than 4cm. It must be pointed out that in the first step of our method, we just use the feature points' coordinates in WCCS as the initial value, and it converges in all the cases.

Table 1 The actual pose of camera recovered by the algorithm (cm)(er. – estimation position error)

No	ideal point	Our Method		Method in [4]	
		estimated point	er.	estimated point	er.
1	(-400,97,84.5)	(-404.1,95.1,83.5)	3.1	(-371.2,98.1,78.7)	28.5
2	(-500,47,84.5)	(-501.7,25.0,97.6)	2.4	(-517.4,35.9, 81.8)	15.8
3	(-500,-103,84.5)	(-498.6,-114.0,91.0)	2.0	(-493.9,-115.2,87.0)	10.3
4	(-400,-3,84.5)	(-403.8,0.3,77.4)	2.3	(-399.1,4.2,85.6)	0.6
5	(-300,-103,84.5)	(-299.2,-104.2,88.1)	0.6	(-323.7,-105.6,82.3)	22.0
6	(-300,-153,84.5)	(-301.2,-150.7,84.0)	0.1	(-291.6,-143.9,85.3)	11.0

In table 1, we also showed the results from our implementation of the method described in [4]. We observe that the method is quite sensitive to appropriate initial value for Newton's method. By using the result of "our method" perturbed by a small amount of noise as the initial value for the method in [4], it is possible to achieve convergence.

5.2 Underwater experiments

Experiments were carried out in the swimming pool of Dalhousie University. We calibrated the camera in the water using a standard calibration pattern. We then used the calibration pattern as our artificial landmark set, and we solved the camera localization problem for several camera posi-

tions (Fig. 7).

Table 2 The localization of the robot by this algorithm (cm)(er. – estimation position error)

No	ideal translation	est. translation	er.
1	(-332, -10.0, 56.8)	(-325.3, -9.5, 43.2)	8.7
2	(-332, 17.0, 56.8)	(-326.4, 13.8, 35.9)	8.6
3	(-332, 44.0, 56.8)	(-328.9, 38.5, 43.1)	5.7
4	(-332, 126.5, 56.8)	(-329.7, 112.5, 50.3)	7.8
5	(-332, 209.0, 56.8)	(-328.9, 198.3, 47.6)	9.4
6	(-332, 291.5, 56.8)	(-326.8, 281.3, 46.8)	11.2

The actual camera position and the estimated camera position are displayed in table 2. The estimation error in the experiment is the distance between the ideal position and the estimated position.

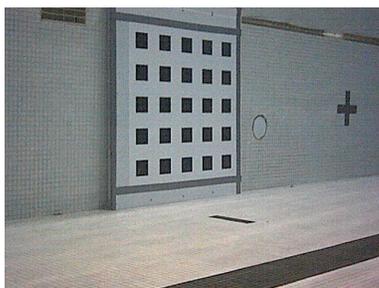


Figure 7 Underwater Image

6 Summary

We have presented a new method for computing the camera location with as few as three point correspondences. After the feature points are recovered in the CCCS with non-linear method, a closed-form solution for the Euclidean transformation between the WCCS and the CCCS can be obtained by geometric transformation. We demonstrated the performance of the algorithm in both the air and underwater. This approach is more reliable and efficient than previous approaches.

Acknowledgements The authors would like to thank Hui Liu, Yongzhen Zhang, Zheyuan Yu, and Gang Wei for help in collecting image data for our experiments.

References

- [1] S. Carlsson. Duality of reconstruction and positioning from projective views. In *Proc. IEEE Workshop on Representation of Visual Scenes*, pages 85–92, 24 June 1995.
- [2] O. Faugeras. *Three-Dimensional Computer Vision, A Geometric viewpoint*. The MIT Press, 1993.
- [3] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24, 1981.
- [4] R. K. P. Horn. *Robot Vision*. The MIT Press, 1986.
- [5] G. Jang, S. Kim, W. Lee, and I. Kweon. Robust self-localization of mobile robots using artificial and natural landmarks. In *IEEE Int. Symposium on Computational Intelligence in Robotics and Automation*, pages 412–417, Kobe, Japan, July 2003.
- [6] R. Kumar and A. Hanson. Robust estimation of camera location and orientation from noisy data having outliers. In *Proc. workshop on Interpretation of 3D Scenes*, pages 52–60, Nov. 1989.
- [7] H. Liu. Acoustic positioning using multiple microphone arrays. In *MS Thesis*, Faculty of Computer Science, Dalhousie University, June 2003.
- [8] Y. Liu, T. S. Huang, and O. D. Faugeras. Determination of camera location from 2D to 3D line and point correspondences. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 12:28–37, 1990.
- [9] H. Madjidi and S. Negahdaripour. Global alignment of sensor positions with noisy motion measurements. In *IEEE Int. Conference on Robotics and Automation*, New Orleans, LA, April 2004.
- [10] G. Roth. Computing camera positions from a multi-camera head. In *Proc. of Int. Conf. on 3-D Digital Imaging and Modeling*, pages 135–142, June 2001.
- [11] S. Se, D. Lowe, and J. Little. Global localization using distinctive visual features. In *IEEE/RSJ Int. Conference in Intelligent Robots and Systems*, pages 227–231, Lausanne, Switzerland, Oct. 2002.
- [12] P. Zhang, C. Zhang, and F. Cheng. Constrained shape scaling of multi-surface objects. In *IEEE on Proc. of the Geometric Modeling & Processing Conference*, pages 398–407, Hong Kong, April 2000.