Robust Camera Calibration for an Autonomous Underwater Vehicle

Matthew Bryant[†], David Wettergreen[‡*], Samer Abdallah[†], Alexander Zelinsky[‡]

Robotic Systems Laboratory
[†]Department of Engineering, FEIT
[‡]Department of Systems Engineering, RSISE
The Australian National University
Canberra ACT 0200 Australia
http://wwwsyseng.anu.edu.au/rsl/

Abstract

At the Australian National University we are developing an autonomous underwater vehicle for underwater exploration and inspection. One of our aims is to track the relative position of underwater targets. This has required the development of a camera calibration system that can overcome the difficulties of underwater vision to provide accurate camera parameters. Conventional camera calibration systems detect and then identify points in an image of a known 3-D calibration pattern. Point identification algorithms typically require the full set of calibration points to be detected to register the target, but this requirement is seldom satisfied in underwater images. We describe a point identification algorithm which does not rely on complete point detection, based upon the indexing of planar invariants calculated from points on a 3-D calibration pattern. Underwater experiments have shown our new method improves the likelihood of successful calibration by up to 80%, and that our calibration system calculates camera parameters enabling range estimation of targets up to 3 metres away with 95% accuracy.

1 Introduction

At the Australian National University we are developing an autonomous underwater vehicle (AUV) named Kambara. We are researching technologies allowing Kambara to search and navigate underwater environments, using visual guidance based on *range estimation* and *feature tracking* techniques.

1.1 Underwater Stereo Camera Calibration

Most range estimation algorithms require knowledge of how pixel coordinates in stereo images correspond to point coordinates in 3-D space. *Camera parameters* define this correspondence, and the procedure of determining the parameters is known as *camera calibration.*

Camera calibration has been an area of extensive research. Calibration algorithms vary across different applications, but most involve processing images of a *calibration pattern*. The camera parameters are calculated from information extracted from the correspondence between points on the pattern, and pixels in the pattern image [Trucco,1998].

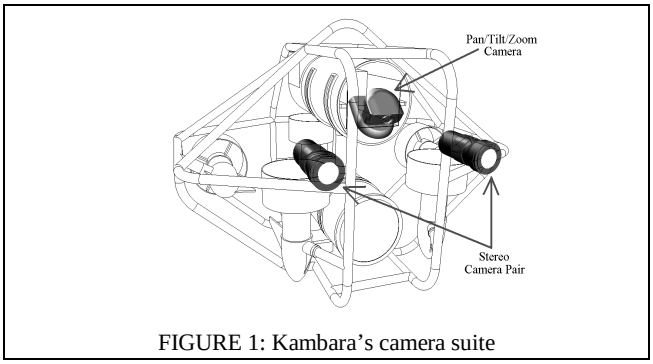

FIGURE 1: Kambara's camera suite

Calibration patterns are required to provide a set of 3-D points, with a relative geometry known to an accuracy that exceeds the required accuracy of the vision system. A typical calibration pattern consists of one or two planar grids of rectangles, or *boxes*, on a contrasting background [Tsai, 1987]. Points are extracted from images of the pattern using edge detection, line fitting, and line intersection techniques. A *point identification algorithm* is applied so that each detected image point can be matched with its corresponding point in 3-D space. If all the points on the calibration pattern are successfully detected then this requires a simple point ordering algorithm.

Underwater environments present challenges to the reliability and accuracy of calibration algorithms. Edge detection is challenged by the fact that the contrast in underwater images reduces with depth [Reynolds, 1998]. Edges in some regions of an image may therefore be missed, and consequently not all points in a calibration pattern image can be reliably detected. Simple point ordering algorithms are therefore impractical in underwater environments.

We have addressed this problem by developing a point identification algorithm which does not rely on the full set of points being detected. The algorithm is based on indexing *planar projective invariants* calculated from points on the calibration pattern.

[*] Now with The Robotics Institute, Carnegie Mellon University.

1.2 Kambara's Vision System

Kambara's vision hardware is comprised of three cameras, shown in Figure 1, and a digitizer. Stereo vision is accomplished through two Pulnix wide angle lens cameras (TMC-73M), housed in moveable waterproof containers mounted on the Kambara frame. A Sony pan-tilt-zoom camera (EVI-D30) is mounted in the upper watertight enclosure, capturing images for Kambara's user-interface. A PXC200 Imagenation framegrabber multiplexes the three camera signals, delivering image frames to the onboard computer.

The following section describes the standard approach to camera calibration, and the problems posed by underwater imaging. Section 3 discusses our new approach to the point identification problem, and section 4 describes our experiments evaluating the robustness of the new approach. Section 5 discusses an experiment testing the underwater accuracy of our calibration system.

2 Camera Calibration Overview

2.1 Camera Parameters

Camera parameters characterise the mathematical model used to describe a camera. Tsai's camera model [Tsai, 1987] uses two parameter categories:

- *extrinsic parameters* define the position and orientation of the camera relative to a world reference frame;
- *intrinsic parameters* define the internal projective geometry of the camera, including focal length and lens distortion.

Appendix A lists and explains these parameters.

2.2 Calibration Algorithm

Most camera calibration algorithms use a calibration pattern to accurately locate points in space [Trucco, 1998 and the references therein]. These 3-D point coordinates are matched with their corresponding image pixel coordinates as the basis for an algorithm which calculates the camera parameters.

A typical calibration pattern uses two planar grids of boxes on a contrasting background [Tsai, 1987], such as the one shown in Figure 2. Algorithm 1 lists the standard algorithm, used with planar box-grid calibration patterns, for finding the parameters of a single camera. The extrinsic parameters of each camera can then be used to derive the stereo extrinsic parameters of the camera suite.

1) Capture an image of the calibration pattern
2) Detect points:
    a) Detect edges
    b) Find box edges
    c) Fit lines to each box edge
    d) Intersect lines to find corner points
3) Identify points
4) Calculate parameters

ALGORITHM 1: A standard camera calibration algorithm.

The performance of a calibration algorithm is judged by the *accuracy* of the camera parameters determined in the parameter calculation step, and the *reliability* of successful calibration. Both performance characteristics are dependent on the quality of the captured image.

2.3 Point Detection

Underwater environments tend to cause errors in edge detection, which propagate through to errors in point detection.

Edge detection locates pixels in regions where the image intensity undergoes sharp variations [Trucco, 1998]. Underwater environments compromise the detection of intensity variations, because the scattering of light by suspended underwater particles decreases image contrast [Reynolds, 1998]. Consequently edges, and therefore corner points, may not be detected in some regions of an image.

The light scattering effect also tends to blur images [Reynolds, 1998], reducing the sharpness in detected variations. This makes it difficult for edge detection algorithms to locate edges accurately. Lines fitted to inaccurately located edges will deviate from the true box edge, leading to inaccurately located corner points.

2.4 Point Identification

Point identification is the task of identifying each detected point in a projected image of the calibration pattern. If all the corner points are detected, as is the case in typical calibration environments, then simple point sorting algorithms can be devised to identify the points on the basis of their vertical and horizontal ordering.

In underwater applications the detection of all points cannot be relied upon, and different techniques must be devised. These are challenged by the fact that much of the geometric information relating the points is lost under projection. In particular, the relative distance between points, the area of a box, and parallelism are not preserved under projective transformations [Mundy and Zisserman, 1992]. We have addressed the problem with an identification scheme based on the indexing of planar projective invariants.

3 Point Identification using Invariant Indexing

We propose a scheme using an index of planar projective invariants to identify boxes in images of a calibration pattern. The identified boxes can then be used to identify the box corner points.

3.1 Kambara's Calibration System

The calibration system developed for Kambara is based upon Algorithm 1, and uses the standard calibration pattern shown in Figure 2. Both planes of the pattern contain 12 boxes, each with dimensions 60mm x 75mm, providing a total of 96 points [Fitzgerald, 1999].

The calibration pattern images captured by the stereo cameras have a 640 x 480 resolution. A Canny edge detection algorithm [Canny, 1996] is used to extract the edges of each box in the image, followed by orthogonal regression line fitting. Lines are then intersected to extract points at the corners of each box.

The point identification step of Algorithm 1 was initially implemented with simple point ordering algorithms [Fitzgerald, 1999], assuming that all the corner points can be detected reliably. This assumption was found to be unrealistic, motivating the development of our new algorithm.

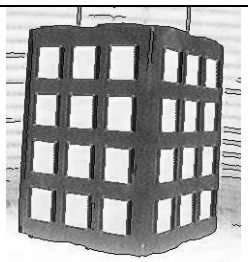

FIGURE 2: Kambara's calibration pattern

3.2 Planar Projective Invariants

A projective invariant is an algebraic property of a set of points which remains constant under projection. Two projective invariants can be calculated for 5 coplanar points x_i, $i \in \{1...5\}$:

$$I_1 = \frac{\left| \begin{bmatrix} x_4 & x_2 & x_1 \end{bmatrix} \right| \left| \begin{bmatrix} x_5 & x_3 & x_2 \end{bmatrix} \right|}{\left| \begin{bmatrix} x_4 & x_3 & x_2 \end{bmatrix} \right| \left| \begin{bmatrix} x_5 & x_2 & x_1 \end{bmatrix} \right|} \quad (1a)$$

$$I_2 = \frac{\left| \begin{bmatrix} x_4 & x_2 & x_1 \end{bmatrix} \right| \left| \begin{bmatrix} x_5 & x_3 & x_1 \end{bmatrix} \right|}{\left| \begin{bmatrix} x_4 & x_3 & x_1 \end{bmatrix} \right| \left| \begin{bmatrix} x_5 & x_2 & x_1 \end{bmatrix} \right|} \quad (1b)$$

where the points x_i are represented as homogeneous column vectors [Rothwell,1995].

There are two important collinearity properties of planar projective invariants which affect the choice and ordering of planar points [Rothwell, 1995]:

1) If any 3 points are collinear, then *either* I_1 or I_2 is singular;
2) If both x_1 and x_2 are collinear with any of the other 3 points, then *both* I_1 and I_2 are singular.

These properties demonstrate the invariants' dependence on point ordering. The second property leads to a constraint where any set of 5 points must be chosen such that x_1 and x_2 are not collinear with a third point.

3.3 Identifying Box-Pairs with Planar Projective Invariants

The invariants given in (1a) and (1b) can be used to identify pairs of boxes. Consider two boxes B_1 and B_2: we can take x_1 and x_2 from B_1, and the remaining points from B_2 to calculate I_1 and I_2. The selection of points used in our algorithm is shown in Figure 3, with the point order chosen to avoid having both I_1 and I_2 singular for any possible combination of boxes.

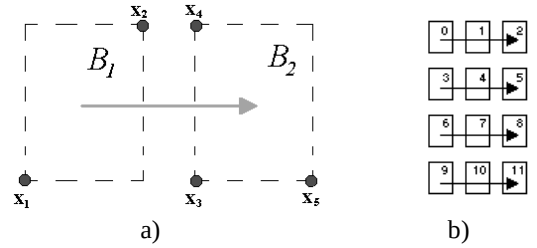

FIGURE 3: a) The corners selected for invariant calculations; b) Box-pairs identical by translation share the same invariant: $I_{0\text{-}2} = I_{3\text{-}5} = I_{6\text{-}8} = I_{9\text{-}11}$.

Two invariants I_1 and I_2 calculated from boxes B_1 and B_2 can be collectively termed as a *box-pair invariant* $I_{B1\text{-}B2}$, graphically represented in Figure 3 as an arrow from box B_1 to box B_2. Each possible box-pair in a calibration pattern will have an associated box-pair invariant.

Box-pair invariants have the following uniqueness properties:

1) For any two boxes B_1 and B_2, $I_{B1\text{-}B2} \neq I_{B2\text{-}B1}$;
2) Box pairs identical under translation share the same invariant.

Figure 3(b) illustrates these uniqueness properties. Numbering the boxes 0 through 11 for one plane of the calibration pattern, we see box-pairs 0-2, 3-5, 6-8, and 9-11 all share the same box-pair invariant.

3.4 Invariant Indexing

A box-pair invariant index can be constructed to aid the identification of box-pairs in images of the calibration pattern. Given a box-pair invariant the index returns a list of the corresponding box-pairs. The index has an entry for each unique box-pair invariant. For a 4 x 3 grid of boxes there are 34 unique invariants associated with 132 box-pair combinations.

The invariants in the index are calculated from precise measurements of the calibration pattern. The invariants used to key the index are calculated from images of the calibration pattern, and will always differ slightly from the index invariants. There are two main causes of this difference:

1) Projective invariants are based on an assumption of pinhole projection. Radial lens distortion in the cameras means the pinhole model is not an accurate model for projection;
2) The edge detection and line fitting of boxes has inaccuracies which propagate through to box corner point calculations, which in turn leads to errors in invariant calculations.

This disparity requires the use of invariant tolerances. When searching the index, a match is found if the image-based invariant falls within the tolerance range of the index invariants. If the tolerance is too small then there will be many false negatives, resulting in many points not being identified. There are more serious consequences if the tolerance is too big: false positives will be made, corrupting the calculation of the camera parameters.

A tolerance of ±20% was empirically determined to work reliably without false positives, and with only a small number of false negatives.

3.5 Index Search Algorithm

The invariant index is used to identify a set of planar boxes. Keying the invariant index with an invariant calculated from two of these boxes will return a list of possible box-pairs. Only one of these is the true box-pair. Calculating invariants between the other boxes can help eliminate possibilities, until a unique combination of boxes has been identified.

This is the basis for the invariant indexing algorithm stated in Algorithm 2. The algorithm uses a search structure, with branches of the structure representing possible box combination scenarios. As more invariants are calculated, branches from the search structure are pruned as possible scenarios are eliminated, until one branch is left which

identifies the boxes.

The index searching algorithm is illustrated in Figure 5. In this example only 4 boxes have been detected from a planar grid: B_1, B_2, B_3, and B_4. The box-pair invariant $I_{B1\text{-}B2}$ between boxes B_1 and B_2 is calculated and used to search the invariant index. A match is found, finding B_1 and B_2 to be one of 3 possible box-pairs. A search structure is created by allocating a branch to represent each of the 3 scenarios: i) B_1 is 3, ii) B_1 is 6, and iii) B_1 is 9.

1) Begin with a list of n unidentified boxes, B_1, B_2....B_n

2) Calculate the invariant $I_{B1\text{-}B2}$ from B_1 to B_2.

3) Key the invariant index with $I_{B1\text{-}B2}$ to find the corresponding list of box-pairs.

4) Create a search structure with one branch allocated to each of the box pairs found from the index.

5) For the remaining boxes B_i, $i \in \{3...n\}$:

   a) Calculate the invariant $I_{B1\text{-}Bi}$
   b) Use $I_{B1\text{-}Bi}$ to look up the invariant index
   c) For each box-pair BP_j, $j \in \{1...k\}$, found in the index:

      i) Search the structure for a branch having the same base box as BP_j.
      ii) If such a branch is found, then add BP_j to it

   d) Prune any branch which doesn't have a base box corresponding to any of the box-pairs.

6) Succeed if only 1 branch remains in the search structure, otherwise the boxes B_1.....B_n are insufficient for identification.

ALGORITHM 2: Identifying boxes using a search structure and an invariant index.

Next the invariant $I_{B1\text{-}B3}$ between boxes B_1 and B_3 is calculated. Searching the invariant index finds that B_1 and B_3 could be one of 6 possible box-pairs. Only two of them share the same base boxes with branches of the search tree, namely box-pairs 6-0 and 9-3. These are added to the corresponding branches. No possible box-pairs are found consistent with scenario i), so it is pruned.

Next the invariant $I_{B1\text{-}B4}$ between boxes B_1 and B_4 is calculated. Searching the invariant index finds that B_1 and B_4 could be one of 6 possible box-pairs. Only 1 box-pair shares the same base box with a branch in the search tree, namely 6-10, and this is added to the corresponding branch. None of the box-pairs are found to be consistent with scenario iii), so this branch is pruned.

Only one branch remains, revealing the true identity of the boxes: B_1 is 6, B_2 is 5, B_3 is 0, and B_4 is 10.

3.6 Performance Limitations

A requirement that must be satisfied for the invariant algorithm to work is that at least one box from each outer boundary of the planar grid must be detected. The number of detected boxes required for identification is therefore dependent on which boxes are detected. A box on two opposite grid corners is sufficient for identification, since each corner box shares two boundaries. The maximum number of boxes required in a 4 x 3 grid is 10.

Radial distortion prevents the target being placed too close to the cameras, as the true invariants increasingly deviate from the pin-hole model invariants. This conflicts with the aim of accurately characterising radial distortion, because the more distortion captured by the image points the better the accuracy of that parameter estimate.

Fortunately this is not a serious problem underwater, since the air-water interface refracts light so as to reduce lens distortion. With Kambara's vision system it was found the calibration pattern can be brought close enough to occupy most of the vertical field of view (FOV) of each camera.

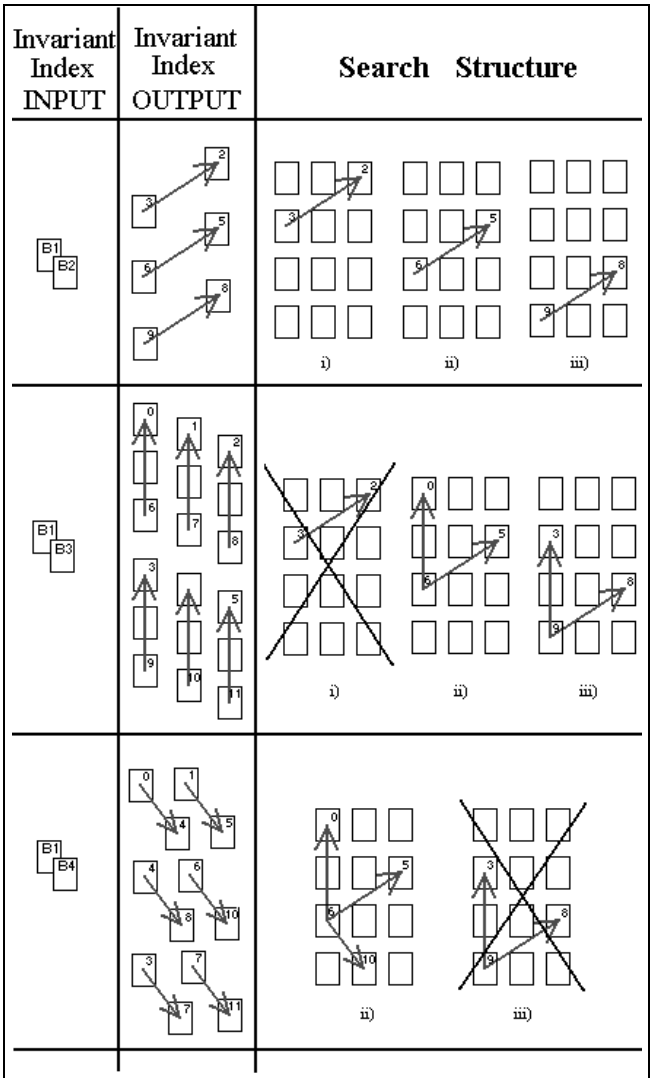

FIGURE 5: An example illustrating Algorithm 2. Boxes B_1, B_2, B_3 and B_4 have been detected in the image of a calibration pattern, but their identity is unknown. The invariant index is used together with the search tree to identify the boxes. The search tree contains branches representing different possible identities. Arrows represent box-pair invariants, while a cross represents a pruned branch of the search structure.

4 Calibration Robustness Evaluation

The performance of the calibration system is judged on both the reliability of the system and the accuracy of the calculated camera parameters. This section describes the experimental evaluation of our calibration system's reliability.

4.1 Experimental Procedure

The performance of the invariant algorithm was tested by comparing the reliability of Kambara's calibration system using a) the invariant indexing algorithm, and b) simple point ordering algorithms reliant on all 96 corner points being detected.

Calibration testing used underwater stereo image sets of the calibration pattern. The image sets were collected by placing Kambara underwater, and capturing images from the stereo cameras as the calibration pattern was moved about the stereo FOV. The stereo camera pair, apart from being slightly verged to enable a close stereo range, were arbitrarily oriented. The calibration pattern was oriented so as to be approximately horizontal in the FOV of each camera.

Not all the captured image sets were fit for calibration. Image sets were discarded if a) both planes of the calibration pattern was not seen clearly in the FOV of both cameras; or b) the calibration pattern was more than 2 metres away from the cameras — if the calibration pattern is further away than this, then each projected box edge has too few pixels for accurate line fitting.

A total of 43 image sets satisfying these requirements were used for testing our calibration technique. Each image set was used twice for calibration, once each for the point ordering and invariant indexing algorithms.

4.2 Results

We found the invariant indexing algorithm to be far superior than simple point ordering. The results of comparison are summarised in Table 1.

Although the point ordering algorithm led to the successful calibration of *one* of the stereo cameras for 40% of the image sets, calibration of *both* cameras never occurred. This result emphasises the difficulty of detecting all 96 corner points in an underwater environment.

The invariant indexing algorithm led to the successful calibration of both cameras for 80% of the image sets.

Point Identification Algorithm	**% of images sets for which at least 1 camera calibrated successfully**	**% of images sets for which BOTH cameras' calibrated successfully**
Point ordering	40%	0%
Invariant indexing	81%	80%

TABLE 1: Evaluation of calibration system reliability with point ordering and invariant indexing point identification algorithms.

5 Calibration Accuracy Evaluation

Underwater environments present challenges to the accuracy of point detection, and consequently the accuracy of camera calibration. This section discusses the challenges of evaluating calibration accuracy, and outlines a methodology which evaluates accuracy using 3-D object dimension estimation.

5.1 Difficulties in Parameter Accuracy Evaluation

Evaluation of the calibration system accuracy is challenging because of the difficulty in obtaining ground truth for comparison.

A camera's intrinsic parameters are dependent on many factors, including the calibration environment, so it is impossible for a camera manufacturer to provide comprehensive parameter specifications.

A camera's extrinsic parameters can in principle be compared with measurements made between camera and world reference frames. In practice, however, such measurements are inaccurate and therefore inappropriate for comparison, since a) the origin of the camera reference frame is located within the camera, and therefore inaccessible to 'ruler' measurements, and b) it is impossible to accurately locate the origin of a camera reference frame.

Another approach might be to use the accuracy of range estimation to infer the accuracy of the camera parameters. Range estimation estimates the vector from a camera reference frame to a point in 3-D space. Unfortunately inaccessibility of the camera reference frame again prevents meaningful comparison with measured distances.

5.2 Experimental Methodology

We have devised an experimental approach which compares the estimated length of objects in Kambara's environment with accurate ruler measurements. The length estimates are found by firstly using range estimation to determine the range vectors to two points in space, and then using simple vector arithmetic to find the distance between the two points. If length estimates are accurate then it can be inferred that range vectors are accurate, which in turn infers that the calculated camera parameters are accurate.

The range estimation algorithm used for this experiment is listed in Appendix B.

5.3 Experimental Procedure

Kambara was placed underwater together with the calibration pattern, and the stereo cameras were calibrated.

The calibration pattern was then used to provide pairs of points separated by precisely known distances. Estimates of point-pair displacements were taken with the calibration pattern approximately 1, 2 and 3 metres away from Kambara. At each of these distances the pattern was moved about the FOV of each camera, and at each position roughly 8 distance measurements between point pairs were calculated.

5.4 Results

The accuracy of length estimation at each distance was calculated as a percentage error of the true (measured) length. The average and maximum percentage errors for each distance are listed in Table 2.

Length estimates were found to have a maximum mean error of 5%, indicating that the calculated camera parameters are sufficiently accurate for range estimation of targets up to 3 metres away. It was also found that range estimates improve as the target approaches closer to the AUV, which is consistent with our intuition that small errors in a range vector's orientation will amplify with distance.

Approximate distance between AUV and target	Mean Error %	Std Dev %	Max Error %
1 metre	1.4 %	1%	6%
2 metre	3.6 %	2.7%	15%
3 metre	5.0 %	5.2%	30%

TABLE 2: Evaluation of calibration system accuracy

6 Conclusion

We have developed a new point identification algorithm, based on planar projective invariant indexing. Experimental evaluation has shown the algorithm to be up to 80% more reliable than simple point ordering algorithms. Further experiments have found Kambara's calibration system, based on our invariant indexing algorithm, to be sufficiently accurate for range estimation applications with target distances up to 3 metres.

There is scope for improvement in the reliability of underwater calibration. Finding an optimum invariant tolerancing scheme would help reduce index false negatives, and therefore increase the calibration success rate. Robustness could be further enhanced by developing techniques allowing the calibration pattern to be held at arbitrary orientations with respect to the camera suite.

In conclusion, we believe that we have developed and tested a camera calibration scheme suitable for underwater computer vision applications.

Acknowledgments

We wish to thank all current and past members of the Kambara team for their tireless support and encouragement. We are especially grateful to Ian Fitzgerald for developing the original Kambara calibration system.

A Camera Parameters

The *intrinsic* parameters of a camera consist of:

- f - focal length of the camera;
- K_1 - radial lens distortion coefficient;
- (C_x, C_y) - pixel coordinates of the centre of lens distortion;
- s_x - horizontal scale factor;
- (d_x, d_y)- the "centre-to-centre" distance between adjacent CCD sensor elements in the X and Y directions;
- N_{cx} - the number of CCD sensors in the X direction;
- N_{fx} - the number of pixels sampled in the X direction.

The *stereo extrinsic* parameters define the mapping between the camera reference frames of the left and right cameras (denoted *{L}* and *{R}*). They consist of a translation vector ${}^{L}P_R = \left[{}^{L}_{R}p_x \;\; {}^{L}_{R}p_y \;\; {}^{L}_{R}p_z \right]^T$ and rotation matrix ${}^{L}_{R}R$.

B Range Estimation Algorithm

The following range estimation algorithm is adapted from the work of [Horn, 1986] and [Tsai, 1987]. The algorithm finds range vectors from the reference frames of the left and right camera to a point *P:* ${}^{L}P = \left[{}^{L}p_x \;\; {}^{L}p_y \;\; {}^{L}p_z \right]^T$ and ${}^{R}P = \left[{}^{R}p_x \;\; {}^{R}p_y \;\; {}^{R}p_z \right]^T$, as illustrated in Figure 6. There are two main steps involved:

1) Use the intrinsic camera parameters for each camera to map the target coordinates (X_f, Y_f) observed in the captured image, to undistorted coordinates (x', y') corresponding to the pinhole camera model [Tsai, 1987]:

$$x' = K_1 X_d^3 + K_1 X_d Y_d^2 + X_d \quad (2a)$$

$$y' = K_1 Y_d^3 + K_1 Y_d X_d^2 + Y_d \quad (2b)$$

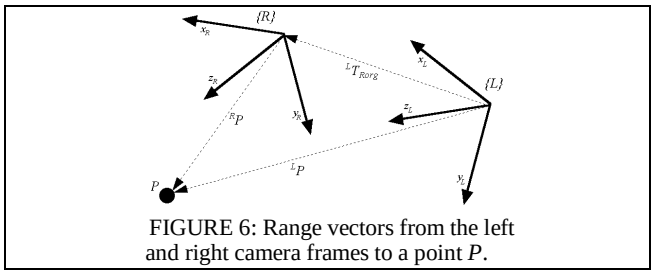

FIGURE 6: Range vectors from the left and right camera frames to a point *P*.

where (X_d, Y_d) are found from (X_f, Y_f) by:

$$X_d = \left(\left(X_f - C_x \right) / s_x N_{fx} \right) N_{cx} d_x \quad (3a)$$

$$Y_d = \left(Y_f - C_y \right) d_y \quad (3b)$$

2) The undistorted image coordinates in the left and right stereo images, (x'_L, y'_L) and (x'_R, y'_R), the focal lengths of each camera, and the stereo extrinsic parameters, are substituted into the following equation, derived from the frame transformations between the stereo camera and the target reference frames [Horn, 1996]:

$$ {}^{L}_{R}R \begin{bmatrix} x'_R / f_R \\ y'_R / f_R \\ 1 \end{bmatrix} {}^{R}p_z = \begin{bmatrix} x'_L / f_L \\ y'_L / f_L \\ 1 \end{bmatrix} {}^{L}p_z \quad (4)$$

This vector equation is solved to find ${}^{L}p_z$ and ${}^{R}p_z$. We then substitute into the following equations to find the range vectors from the left and right cameras, ${}^{L}P$ and ${}^{R}P$:

$${}^{L}P = \begin{bmatrix} \frac{x'_L}{f_L} & \frac{y'_L}{f_L} & 1 \end{bmatrix}^T {}^{L}p_z \quad (5a)$$

$${}^{R}P = \begin{bmatrix} \frac{x'_R}{f_R} & \frac{y'_R}{f_R} & 1 \end{bmatrix}^T {}^{R}p_z \quad (5b)$$

References

[Canny, 1996] Canny, J. *A Computational Approach to Edge Detection*. IEEE Trans. on PAMI, Vol. 8, No. 6, pp. 679-698.

[Fitzgerald, 1999] Fitzgerald, I. *A Vision System for an Autonomous Underwater Vehicle*. Department of Engineering ANU, 1998.

[Horn, 1996] Horn, B.K.P. *Robot Vision*. The MIT Press Cambridge, Massachusetts, 1996.

[Mundy and Zisserman, 1992] Mundy, J. and Zisserman, A. *Geometric Invariance in Computer Vision*. MIT Press, 1992.

[Reynolds, 1998] Reynolds, J. *Autonomous Underwater Vehicle: Vision System*. Department of Engineering ANU, 1998.

[Rothwell, 1995] Rothwell, C. A. *Object Recognition Through Invariant Indexing*. Oxford University Press, New York, 1995.

[Trucco, 1998] Trucco, E. and Verri, A. *Introductory Techniques for 3-D Computer Vision*. Prentice-Hall, 1998.

[Tsai, 1987] Tsai, R. Y. *A Versatile Camera Calibration Technique for High Accuracy 3D Machine Vision Metrology Using Off-the-shelf TV Cameras and Lenses*. IEEE Trans. on Robotics and Automation,Vol. RA-3, No. 4, pp. 323–344, 1987.