Camera Calibration in Wireless Multimedia Sensor Networks

Junbin Liu [#,*], Damien O'Rourke [#], Tim Wark [#,*], Ruan Lakemond [*], Sridha Sridharan [*]

[#] *Autonomous Systems Laboratory, CSIRO ICT Centre*
Brisbane, Australia
{jim.liu, damien.o'rourke, tim.wark}@csiro.au

[*] *SVAIT Laboratory, Queensland University of Technology*
Brisbane, Australia
{junbin.liu, r.lakemond, s.sridharan}@qut.edu.au

***Abstract*—Wireless Multi-media Sensor Networks (WMSNs) have become increasingly popular in recent years, driven in part by the increasing commoditization of small, low-cost CMOS sensors. As such, the challenge of automatically calibrating these types of cameras nodes has become an important research problem, especially for the case when a large quantity of these type of devices are deployed. This paper presents a method for automatically calibrating a wireless camera node with the ability to rotate around one axis. The method involves capturing images as the camera is rotated and computing the homographies between the images. The camera parameters, including focal length, principal point and the angle and axis of rotation can then recovered from two or more homographies. The homography computation algorithm is designed to deal with the limited resources of the wireless sensor and to minimize energy consumption. In this paper, a modified RANdom SAmple Consensus (RANSAC) algorithm is proposed to effectively increase the efficiency and reliability of the calibration procedure.**

***Keywords*-wireless sensor networks; camera calibration; homography;**

I. Introduction

Wireless Multi-media Sensor Networks (WMSNs) are rapidly decreasing in cost while increasing in performance ability in terms of energy efficiency and processing capacity. As a result, this class of device is becoming increasingly popular for surveillance and environmental monitoring applications. If networks of wireless sensors with cameras are to be used for higher level multiple camera surveillance tasks, such as tracking, then they must be calibrated. The features that make WMSNs attractive (low cost, high numbers, ability to deploy to remote locations) additionally make manual calibration difficult and not cost-effective. As a first step towards calibration of camera networks (in terms of multi-camera correpsondances), efficient automatic calibration procedures are thus required for these applications.

This paper presents a method for self calibration of a panning wireless multimedia sensor node, where the node consists of a camera, a servo for panning (rotating around the vertical axis), and a DSP module. The calibration method makes use of the panning ability to capture images at different camera orientations and then establishes the correspondence between the 2 consecutive image pairs. This process cannot be done simply by using the angle of servo rotation due to the inaccuracy of the servos and thus alternative methods must be used. SURF feature extractor and descriptor [1] is employed, due to its robustness and speed, to find corresponding points between the images.

The homographies (projective transformation) between the images are computed using the set of putative correspondences and a modified RANdom SAmple Consensus (RANSAC) [2], [3], [4], [5] algorithm. The intrinsic camera parameters (focal length and principal point) are computed from two or more homographies using a method similar to [6]. The angle of rotation between the images and the axis of rotation (extrinsic parameters) are also recovered.

Two main contributions of this paper are:

- A modified RANSAC algorithm is proposed. It is capable of reducing the processing time and improving the likelihood of finding a good model.
- A practical approach towards self-calibration of WMSN sensors is also presented.

II. Background and Related Work

A. Self Calibration of a Stationary, Rotating Camera

Camera calibration is the process of finding the parameters of a parametric model describing the camera operation. A brief description of the model we use is given below where more details can be found in [3]. We employ the perspective pinhole camera model which may be expressed as follows:

$$\mathbf{x} = \mathbf{M}\mathbf{X} \tag{1}$$

where $\mathbf{X} = [x, y, z, 1]^\top$ is the world coordinates, in projective 3-space ($\mathcal{M}^3$). $\mathbf{x} = [x, y, w]^\top$ is the image coordinate in projective 2-space ($\mathcal{M}^2$), and $\mathbf{M}$ is a 3×4 projection matrix that maps the world coordinates to image coordinates. The matrix $\mathbf{M}$ may be decomposed as,

$$\mathbf{M} = \mathbf{K} \begin{bmatrix} \mathbf{R} & -\mathbf{R}\mathbf{t} \end{bmatrix} \tag{2}$$

Here $\mathbf{R}$ is a rotation matrix representing the camera orientation and $\mathbf{t}$ is a translation vector representing the camera center

point. The matrix $\mathbf{K}$ is referred to as the camera calibration matrix as has the following form:

$$\mathbf{K} = \begin{bmatrix} \gamma f & s & p_u \\ 0 & f & p_v \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

The parameters of this matrix are referred to as the intrinsic camera parameters, and consist of the camera focal length f and principal point (p_u, p_v), as well as aspect ratio γ, which is 1 for most cameras, and the skew s, which is 0 for most cameras.

During self calibration of a stationary camera, the world coordinate system may be chosen such that the camera center is at the origin, so that the translation vector $\mathbf{t} = \mathbf{0}$. The camera model presented in Equation 2 may then be simplified to,

$$\mathbf{M} = \left[\begin{array}{c|c} \mathbf{KR} & \mathbf{0} \end{array} \right] \tag{4}$$

The objective from here is to find the intrinsic calibration parameters ($\mathbf{K}$), and the extrinsic orientation ($\mathbf{R}$).

A review of self calibration literature can be found in [7]. The procedure for calibrating a stationary, rotating camera described in [6] was used in this work, since the cameras used have the ability to rotate around the vertical axis. This method makes use of three images taken from the same point in space at different orientations and does not require any prior knowledge of the camera parameter. An extended method is presented in [8], [9] that is suitable for calibrating a rotating and zooming camera. This may be of use in future, when wireless sensors gain the ability to zoom, but is not used here.

A point $\mathbf{X}$ observed by a camera $\mathbf{M}_i = \mathbf{KR}_i$ produces the imaged point $\mathbf{x}_i = \mathbf{KR}_i\mathbf{X}$. Any two images of this same point taken by cameras $\mathbf{M}_i$ and $\mathbf{M}_j$ (cameras related by a rotation) are related by a homography (projective transformation between two images) $\mathbf{P}_{ij}$,

$$\mathbf{x}_i = \mathbf{P}_{ij}\mathbf{x}_j = \mathbf{KR}_i\mathbf{R}_j^{-1}\mathbf{K}^{-1}\mathbf{x}_j \tag{5}$$

If the coordinate axes are chosen to align with reference camera 0 such that $\mathbf{R}_0 = \mathbf{I}$, then the image of point $\mathbf{X}$ in any image j is related to the reference image according to:

$$\mathbf{x}_j = \mathbf{P}_j\mathbf{x}_0 = \mathbf{KR}_j\mathbf{K}^{-1}\mathbf{x}_0 \tag{6}$$

Rearranging in terms of $\mathbf{R}_j$ gives:

$$\mathbf{R}_j = \mathbf{K}^{-1}\mathbf{P}_j\mathbf{K} \tag{7}$$

Because $\mathbf{R}_j$ is a rotation matrix, it is equal to its inverse transpose, and therefore it follows that,

$$\left(\mathbf{KK}^\top\right)\mathbf{P}_j^{-\top} = \mathbf{P}_j\left(\mathbf{KK}^\top\right) \tag{8}$$

where $\mathbf{KK}^\top$ is a 3×3 symmetric matrix with 6 independent entries.

Equation 8 gives rise to nine equations in the six independent entries in $\mathbf{KK}^\top$. Due to redundancy, at least two homographies are required for a unique solution, giving rise to an over constrained system that may be solved using least squares. Once $\mathbf{KK}^\top$ is found, $\mathbf{K}$ may be found by means of Choleski factorization and the rotation matrices may be computed using Equation 7.

B. Homography Computation using Local Feature Matching and RANSAC

The calibration method described in the previous section requires two or more homographies between images taken at different camera orientations. The homographies can be computed by iterative image alignment [10] or by matching local image features. The local image features approach is a natural choice for implementation in a WMSN because it requires significantly less computation effort and memory. Local image features are patterns in an image that are distinguishable from the surrounding image. They are matched across views by computing a descriptor for each feature from its local image neighborhood. A comprehensive review of local image feature detectors was recently published in [**?**]. The SURF feature extractor and descriptor [1] was chosen because it is efficient, suitable for implementation using only integer numbers and sufficiently robust for the task. Since the set of putative correspondences produced by matching SURF features generates a significant proportion of incorrect matches (outliers), the commonly used RANdom SAmple Consensus (RANSAC) [2] was chosen for estimating homographies [3].

The basic idea of RANSAC is to randomly select the smallest subset of data points (four pairs of correspondences in the case of homography computation), compute the model from this subset and then see how many of the available data points agree with this model. This is repeated many times in order to try and find the model that agrees with the largest number of data points. Samples are selected at random because it is usually not feasible to evaluate every possible combination of data points. The number of trials required to ensure a good likelihood of finding the best solution depends on the number of expected outliers. The following formulation, proposed in [2], can be used to find the number of trials N required to achieve a probability z of obtaining the correct solution, given that p points are required per trial ($p = 4$ in this case) and the data is expected to contain w proportion inliers:

$$N = \frac{\log\left(1 - z\right)}{\log\left(1 - w^p\right)}. \tag{9}$$

The generic RANSAC algorithm has been extended by various authors, for example [4] introduces bias in the sample selection probability based on the estimated reliability of features and computes a model quality measure based on inlier error; [5] achieves speed improvements to the model validation stage by introducing a pre-validation check; [11] attempts to compute the probabilities of the validities of the correspondences and uses this information to accelerate the process; [4] introduces a geometric constraint to remove outliers *before* applying RANSAC. To the authors' knowledge, no published work is exists on modifying the random point selection process to improve the accuracy of the homographies. We present our method for addressing this problem in the following section.

III. An Improved RANSAC Algorithm for Homography Computation

It has been observed that the generic RANSAC method produces inaccurate homographies when the feature correspondences occupy a small section of the region of overlap between images. This is primarily due to the fact that the algorithm selects four pairs from a pool of putative correspondences based on a uniform random process. Each pair of interest points has an equal chance of being selected for computing the homography. This implies that if the four points were selected in a way that they are clustered (with no three co-linear) near one edge of the image, it is highly probable that points at the opposite edge are mismatched by the calculated homography.

To deal with this limitation, modification to the uniform random point selection process was made in order to improve the accuracy of RANSAC. For a set of tentative pairs c_i, the variance of all points in the first image is estimated. This is done in both the x and y directions giving σ_x^2 and σ_y^2, which indicates the spread of all the putative matches. Now a set of four points can be defined as:

$$\begin{aligned}
\mathbf{p}_1 &: \ (\overline{x} - 2\sigma_x, \overline{y} - 2\sigma_y), \\
\mathbf{p}_2 &: \ (\overline{x} - 2\sigma_x, \overline{y} + 2\sigma_y), \\
\mathbf{p}_3 &: \ (\overline{x} + 2\sigma_x, \overline{y} - 2\sigma_y), \\
\mathbf{p}_4 &: \ (\overline{x} + 2\sigma_x, \overline{y} + 2\sigma_y).
\end{aligned}$$

The first image is then divided into $s_t \times s_t$ square tiles ($s_t = 10$ in this implementation), and each element in c_i is indexed according to the tile in which it falls. A tile selection process starts by sampling a random integer coordinate from a Gaussian distribution with mean at point $\mathbf{p}_i$ and x-axis standard deviation of $width/8$, y-axis standard deviation of $height/8$. This coordinate is then converted into a tile index that indicates the tile from which a putative pair is to be selected. In case of two or more pairs existing in the same tile, a random pair is selected. If there is no putative pair in a particular tile, the tile selection process is repeated. The whole process is repeated for $i \in \{1, 2, 3, 4\}$, selecting a set of four putative pairs and form the subset used to estimate a homography model. This biases the random sample selection process towards the boundaries of the feature cluster resulting in an increased probability of four far-separated points being selected. Figure 1 shows the use of a homography computed by iRANSAC in mapping one image onto another.

Algorithm 1 lists the modified RANSAC algorithm. This algorithm makes use of the following subroutines:

- $(\overline{x}, \overline{y}) \leftarrow$ MEAN(c) computes the mean coordinates of the elements in the set of correspondences belonging to image 1.
- $(\sigma_x^2, \sigma_y^2) \leftarrow$ VARIANCE(c) computes the variance of coordinates of the elements in the set of correspondences belonging to image 1.
- $(i_x, i_y) \leftarrow$ RANDN $(\mathbf{x}, s_t, \sigma_x, \sigma_y)$ samples a random set of integer coordinates from a Gaussian distribution with mean at point $\mathbf{x}$ and x-axis standard deviation of σ_x, y-axis standard deviation of σ_y and zero covariance. The integer coordinates are then converted to tile indices (i_x, i_y) using tile size s_t.

Fig. 1. Mosaic of two images related by a rotation, circles represent matched features

IV. Experimental Evaluation

A. System Overview

The calibration procedure presented in this paper is designed to be run on the wireless camera node developed by CSIRO [12], [13]. The node consists of three layers of devices, which are: camera board, digital signal processor board and a Fleck board. The node is powered by four AA batteries. The camera board contains a replaceable camera lens and the OV9655 high-performance image sensor, developed by OmniVision Technologies. This chip has been selected due to its reasonable resolution (1.3 Mega-pixel) and energy cost ($\approx$ 90 mW). The DSP daughter board is responsible for image processing. It contains a BF537 Blackfin digital signal processor, running at clock rate of 600 MHz. There is 2MB onboard flash available for data storage and 32MB of SDRAM. The Fleck board, consists of an Atmega128 micro-controller running at 8 MHz, a Nordic NRF905 radio transceiver and a 15cm whip antenna with a 500 meter range, capable of sending data at 76.8 Kbits/s. Please refer to [12], [13] for a more detailed description of the hardware platform.

The WMSN node is mounted on top of a high-tech servo[14], which can be controlled to rotate through a series of angles. The horizontal and vertical axis that are parallel to the image plane of the camera is defined to be x and y-axis. z-axis is then the axis that goes through the center and perpendicular to both x and y axis. With this definition, the unit vector of axis of rotation of the camera must be of the form $[0, 1, 0]^T$ or $[0, -1, 0]^T$. Although there has been attempts to aligning the axis of rotation of the servo exactly with the optical center of the camera, this was not as precise as desired due to mechanical problems. The complete node setup is shown in Figure 2.

B. Complete Calibration Procedure

When the cameras are first deployed, the following calibration procedure should take place to recover the focal length and the principal points of the camera:

Function $\{\mathrm{P}_c, c_c\} \leftarrow \text{iRANSAC}\,(c_i, n_t, w_x, w_y)$
Input:
$c_i = \left\{(\mathbf{x}_1, \mathbf{x}_2)_1, (\mathbf{x}_1, \mathbf{x}_2)_2, \ldots, (\mathbf{x}_1, \mathbf{x}_2)_{n_x}\right\}$ – A set of putative correspondences, where n_x is the total number of putative pairs.
w_x, w_y – The image dimensions.
Output:
n_t – The number of trials to run.
c_c – The consensus set – elements from c_i that all fit the selected model.
P_c – The homography fit to the consensus set using least squares.
begin
  $(\overline{x}, \overline{y}) \leftarrow \text{MEAN}(c_i)$.
  $(\sigma_x^2, \sigma_y^2) \leftarrow \text{VARIANCE}(c_i)$.
  $\mathbf{p}_1 \leftarrow (\overline{x} - 2\sigma_x, \overline{y} - 2\sigma_y)$.
  $\mathbf{p}_2 \leftarrow (\overline{x} - 2\sigma_x, \overline{y} + 2\sigma_y)$.
  $\mathbf{p}_3 \leftarrow (\overline{x} + 2\sigma_x, \overline{y} - 2\sigma_y)$.
  $\mathbf{p}_4 \leftarrow (\overline{x} + 2\sigma_x, \overline{y} + 2\sigma_y)$.
  Divide image into $s_t \times s_t$ square tiles.
  Index each element in c_i according to the tile in which the point from image 1 falls.
  $\sigma_x \leftarrow w_x/8$.
  $\sigma_y \leftarrow w_y/8$.
  $c_c \leftarrow \emptyset$.
  $n_c \leftarrow 0$.
  $n_t \leftarrow \infty$.
  while $i < n_t$ **do**
    for $j \in \{1, 2, 3, 4\}$ **do**
      $v \leftarrow false$.
      repeat
        $(i_x, i_y) \leftarrow \text{RANDN}\,(\mathbf{p}_j, s_t, \sigma_x, \sigma_y)$.
        if *tile* (i_x, i_y) *contains correspondences* **then**
          Select a random correspondence from tile (i_x, i_y) and store in $d_t\{j\}$.
          $v \leftarrow true$.
        end
      until v
    end
    Compute the trial model P_t from the trial set d_t.
    Determine the consensus set for this trial c_t by checking which of c_i fit model P_t.
    if *SIZE* $(c_c) > n_c$ **then**
      $c_c \leftarrow c_t$.
      $n_c \leftarrow \text{SIZE}\,(c_t)$.
      $w \leftarrow \text{SIZE}\,(c_t)\,/\text{SIZE}\,(c_i)$.
      $n_t \leftarrow log(1 - z)/log(1 - w^p)$.
      where $z = 0.99$ and $p = 4$.
    end
    $i \leftarrow i + 1$.
  end
  Compute the output model P using least squares fit to consensus set c_c.
end

Algorithm 1: iRANSAC Homography Algorithm with Biased Sample Selection.

Fig. 2. Wireless multimedia node developed by CSIRO

1) Capture three images that are related by pure rotations using sevro-mechanism [12].
2) Extract SURF features from all images.
3) Match features extracted between successive images using Euclidean distance between feature vectors.
4) Compute at least two homographies from two or more sets of feature correspondences using the improved-RANSAC as presented in section III
5) Calculate the focal length and principal point of the camera using Equation 8, as described in section II-A

C. Experimental Setup

Our experimental validation was divided into two main stages. The first stage evaluated the performance gain of the improved RANSAC algorithm (iRANSAC) when compared with the generic RANSAC algorithm (gRANSAC). The second stage evaluated the accuracy of the calibration procedure. These experiments were conducted on four pre-captured data sets, which were obtained by placing the node at various locations to capture VGA quality images. For each set, the servo rotated through a fix angle($5°, 10°$ or $15°$). The names of the data sets are car park (C.P.), lecture theater (L.T.), process bay (P.B.) and driveway (D.W).

1) Comparison of iRANSAC and gRANSAC: Due to the random point selection process of RANSAC, the accuracy of the models estimated can only be reflected in the spread of the measurement when a large number of identical tests are performed. Since the homography itself doesn't reflect any real-world quantities, angle of rotation and the axis of rotation of the camera was chosen for comparsion.

Equation 7 describes the relationship among the homography $\mathbf{P}$, the rotational matrix $\mathbf{R}$ and the calibration matrix $\mathbf{K}$. $\mathbf{K}$ is kept constant so that the accuracy of the homography $\mathbf{P}$ is reflected by $\mathbf{R}$, which can be computed directly from simple matrix operations. The constant $\mathbf{K}$ matrix is formed using a focal length of 766, principal point of $(313, 226)$, aspect ratio of 1 and 0 skew.

$$\mathbf{K} = \begin{pmatrix} 766 & 0 & 313 \\ 0 & 766 & 226 \\ 0 & 0 & 1 \end{pmatrix}.$$

This was obtained offline using the calibration toolbox supplied by CalTech [15] with a checkerboard patter and thus is referred to as the offline calibrated result throughout this paper. Since $\mathbf{R}$ is obtained from Equation 7, the angle of rotation around the axis of rotation and the unit vector that represents the rotational axis are then computed from $\mathbf{R}$. The results are presented in Figure 3, 4 and 5.

2) Calibration Accuracy: The camera parameters, i.e. focal length and principal point are recovered from each of the data sets described above. The procedure followed are outlined in Section IV-B. Again due to the use of RANSAC, the accuracy is not reflected through a particular measurement, but rather the distribution of the results over a large number of identical test runs. The results of calibration are presented in table I.

D. Results and Evaluation

Figure 3 shows histograms of angle of rotation, generated using both iRANSAC and gRANSAC over 5000 test runs for a selected pair of images. Similarly Figure 4 shows the comparison of each of the 3 components of rotational axis, which has an ideal value of $[0, 1, 0]^T$. It is evident that quantities generated using iRANSAC possess a much narrower distribution, indicating a more accurate model. This improvement is a result of selecting four geometrically separated points for computing the model.

The servo was programmed to rotate through an angle of 10° between the images used to generate Figure 3, but the histograms are centered around a value of approximately 8.9°. This is caused by two types of errors: the inaccuracy of the servo and the imperfect alignment of y-axis and the optical center of the camera.

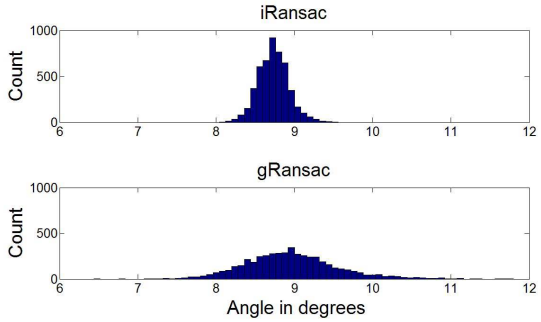


Fig. 3. Histogram of angles calculated over 5000 test runs.

iRANSAC and gRANSAC are further compared using the data sets described above. The algorithm was executed 500 times for each pair of images in each set to estimate the homography and the results were averaged across all image pairs. Figure 5 (a) and (b) shows the comparison of the two algorithms in terms of number of trials and the standard deviation of the angles. It is evident that the modification made to RANSAC brings up to 50% reduction in the processing time and increase in accuracy. The effect of the reducation in number of trials has a more significant effect when the number of trials is high and takes a large percentage of the overall processing time of the calibration procedure.

Table I shows the results of calibration of the same camera with the 4 data sets. Results are averaged from 100 identical test runs for each data set. The last entry of table I is an offline calibration result, calculated using the toolbox provided by CalTech [15] with a checkerboard and manual input. It can be seen that the focal length and the principal point is consistent throughout all the data sets and the principal points match the offline calibrated result but the focal lengths are 6% higher than the offline calibrated result. This consistent bias is caused by the imperfect alignment of the axis of rotation (y-axis) and the optical center of the camera and the geometric distortion of the lens which is not modelled here.

V. Conclusion and Future Work

Self-calibration of camera nodes is a crucial task that must be undertaken if the full capabilities of wireless multimedia sensor networks are to be utilized for tasks such as tracking and image registration. To help address this issue we have presented an improved RANSAC method based on the use biased random sample selection. The results have shown that if the random sample selection process is biased towards the boundaries of the feature cluster, a significant increase in model accuracy and reduction in processing time can be achieved. The accuracy of the calibration is consistent with the offline calibrated result, but relies on the precision of homographies.

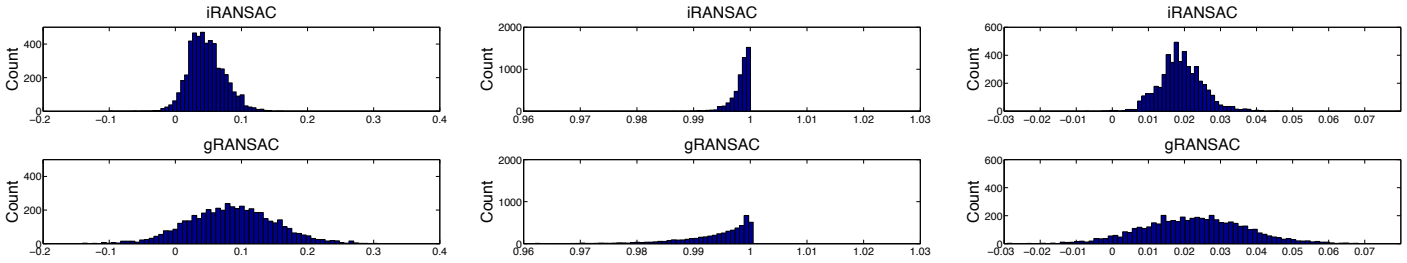
It was also observed that in some outdoor environments, images related by even a small rotation of 5° may often not have many features in common. This is predominantly due to a large proportion of the scene being perpendicular to the image plane of the camera. In addition, these features may be distributed around a certain region and do not occupy the entire region of overlap. This could lead to inaccurate or completely wrong homographies. Furthermore, for outdoor deployments, the cameras have to be able to cope with dynamic scene changes, such as moving tree branches or rapidly changing light conditions. If moving features dominate the region of overlap, it is not viable to compute a correct homography. The problems raised above lead to an additional question regarding the selection of images. Sophisticated modeling methods and selections based on statistical beliefs could be employed to solve this issue and will be studied in the future work.

An extension of current work to multi-camera calibration will be the center of the future work. Algorithms presented in this paper, including feature matching, improved RANSAC, homography computation and self-calibration procedure are step stones towards camera network calibration where the correspondences among multiple cameras are computed.

References

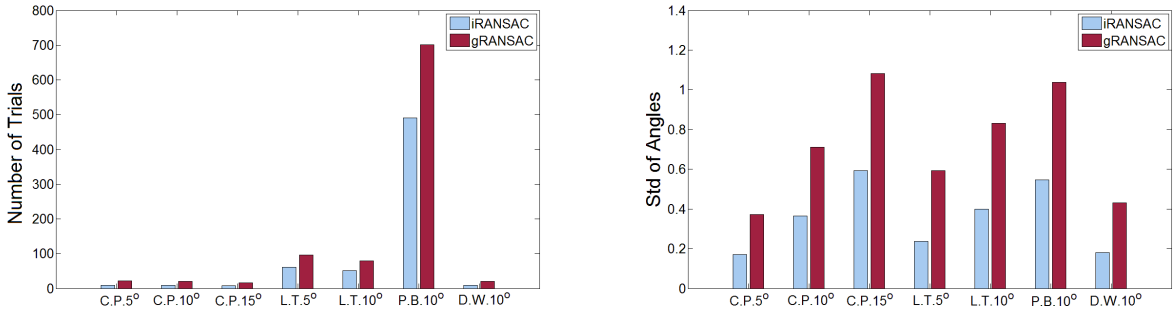
[1] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Surf: Speeded up robust features," *Computer Vision and Image Understanding (CVIU)*, vol. 110, no. 3, pp. 346–359, 2008.

[2] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Comm. of the ACM*, vol. 24, pp. 381–395, 1981.



(a) X component (b) Y component (c) Z component

Fig. 4. Histogram of each of the 3 components of the axis of rotation vector



(a) Number of trials (CP = car park, LT = lecture theater, PB = process bay, DW = driveway)

(b) Standard deviation of angles

Fig. 5. Comparison between improved and generic RANSAC in terms of proportion of inliers, number of RANSAC trials and standard deviation of angles (Car Park (C.P.), Lecture Theater (L.T.), Process Bay (P.B.) and Driveway (D.W))

TABLE I
CALIBRATION RESULTS OVER 100 TEST RUNS ON 4 DATA SETS FOR AN ANGLE OF 10° ONLY.

Loc.	Alg.	Focal length (pixels)	std of focal length	pricipal point	std in principal point
Car Park	iR	812.1	15.2	(317.5, 226.1)	(3.6, 7.8)
	gR	812.1	31.6	(319.5, 232.5)	(7.1, 13.8)
Lect. Th	iR	812.4	11.4	(321.2, 234.6)	(4.3, 9.7)
	gR	827.1	32.9	(323.1, 221.8)	(11.1, 19.8)
Proc. Bay	iR	808.5	8.0	(321.0, 232.1)	(2.2, 4.4)
	gR	802.7	18.2	(321.5, 230.8)	(5.5, 9.7)
Drive way	iR	812.5	14.8	(323.3, 223.7)	(3.6, 6.3)
	gR	814.6	23.6	(324.9, 232.6)	(11.0, 14.9)
Calib. Res.	na	766	na	(313.2, 226.8)	na

[3] R. Hartley and A. Zisserman, *Multiple View Geometry in computer vision*, 2nd ed. Cambridge University Press, 2004.

[4] P. Marquez-Neila, J. Garcia Miro, J. M. Buenaposada, and L. Baumela, "Improving ransac for fast landmark recognition," in *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on*, 2008, pp. 1–8.

[5] O. Chum and J. Matas, "Optimal randomized ransac," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 8, pp. 1472–1482, 2008.

[6] R. I. Hartley, "Self-calibration of stationary cameras," *Int. J. Comput. Vision*, vol. 22, no. 1, pp. 5–23, 1997, 250491.

[7] E. Hemayed, "A survey of camera self-calibration," in *Proceedings. IEEE Conference on Advanced Video and Signal Based Surveillance, 2003.*, July 2003, pp. 351–357.

[8] L. D. Agapito, R. I. Hartley, and E. Hayman, "Linear calibration of a rotating and zooming camera," in *In Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 1999, pp. 15–21.

[9] L. Agapito, E. Hayman, and I. Reid, "Self-calibration of rotating and zooming cameras," *Int. J. Comput. Vision*, vol. 45, no. 2, pp. 107–127, 2001.

[10] S. Baker and I. Matthews, "Lucas-kanade 20 years on: A unifying framework: Part 1," *International Journal of Computer Vision*, vol. 56, no. 3, pp. 221–255, 2004.

[11] B. J. Tordoff and D. W. Murray, "Guided-mlesac: faster image transform estimation by using matching priors," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, no. 10, pp. 1523–1535, 2005.

[12] D. O'Rourke, R. Jurdak, J. Liu, D. Moore, and T. Wark, "On the feasibility of using servo-mechanisms in wireless multimedia sensor network deployments," in *Fourth IEEE International Workshop on Practical Issues in Building Sensor Network Applications*, 2009.

[13] T. Wark, P. Corke, J. Liu, and D. Moore, "Design and evaluation of an image analysis platform for low-power, low-bandwidth camera networks," in *Workshop on Applications, Systems, and Algorithms for Image Sensing 2008*, 2008.

[14] ServoCity. Available from http://www.servocity.com/ [Accessed 28 May 2009].

[15] Camera Calibration Tool Box, CalTech. Available from http://www.vision.caltech.edu/bouguetj/calib_doc/ [Accessed 10 May 2009].