Simplified Vehicle Calibration Using Multilinear Constraints

H. Stewenius

Centre for Mathematical Sciences
Lund Institute of Technology
Box 118
SE-22100 Lund, Sweden
stewe@maths.lth.se

Abstract

An Autonomously Guided Vehicle using both odometry and visual data for navigation needs calibration parameters. These include camera placement as well as parameters relating odometry to vehicle motion. Calibration of these parameters is related to the Hand-Eye calibration problem. Instead of using a calibration target or trying to solve for structure and motion a novel method using the continuous multilinear constraint to test parameter combinations is proposed. A low order polynomial target function is calculated in linear time over the sample size resulting in very fast iterations in the optimisation step.
The method is tested on simulated data and increased sample size improves the parameter estimates.

1 Introduction

This paper describes a novel method for vehicle calibration of an autonomous guided vehicle AGV. The vehicle uses both odometry and visual data in navigation. The general structure of the vehicle is known and motion of the vehicle as well as for the camera can be computed from signals in the vehicle if certain parameters are supplied. The problem to calibrate these parameters is related to Hand-Eye calibration. The problem of self-calibration on an AGV is not of purely academic interest as industrial AGVs are usually custom built for a specific customer by joining a control and camera system to a robot vehicle from different suppliers and it is not in this case certain that all parameters are well known or at least well known to the control unit. The environment where calibration has to be performed is usually unknown. A longer introduction to one such system can be found in [5].
For AGV the previous attempts at calibration has required a known map and the computation of this map either required the map to be known from measuring it by hand or by measuring it with a vehicle with known parameters. Another possibility would be to use the camera data to build a map without odometry information. This is possible but very time consuming as well as not perfectly

J. Bigun and T. Gustavsson (Eds.): SCIA 2003, LNCS 2749, pp. 669–676, 2003.

stable.
In the related field of Hand-Eye calibration three different tracks for computing the Hand-Eye relation are visible. The first of these is to use a well known calibration target [9], the second is to compute structure and motion and the third to compute motion from the motion field, [4]. We use the standard camera projection equation

$$\lambda u = PU, \tag{1}$$

where u is the image point, P camera matrix and U real world coordinates of the point. The camera and vehicle calibration parameters x together with odometry data o_k can be used to calculate the camera motion.

$$\dot{P}(t_k) = \dot{P}(x, o_k).$$

It is possible to get the global motion of the vehicle, but this involves integration of odometry data over time. A typical problem here is that small errors accumulate to produce large errors over time. Odometry is better suited on a short time scale. One could in theory use the ordinary camera equation (1) to find the calibration parameters but this would require solving for the unknown scene points as well.

Using multilinear constraints has several advantages. Firstly, the constraint involve camera motion P and image motion u alone. The projective depth λ and the scene structures U are eliminated. Secondly the constraints involve only relative motion. Thus at each time instant and image point, we are free to use a different coordinate system for the camera motion.

2 Computer Vision and Vehicle Motion

Multilinear constraints

The multilinear constraints tell us if a certain camera movement is coherent with respect to the images from this camera. In order to understand the matching constraints in the case of continuous time, it is necessary to take a look at the corresponding constraints in the discrete time case. For a more thorough treatment, see [3]. Alternative formulations of the same type of constraints can be found in [1, 8]. The first order discrete multilinear constraint in space is equivalent to

$$\det \begin{bmatrix} P_0 & u_1 & \mathbf{0} \\ P_1 & \mathbf{0} & u_1 \end{bmatrix} = 0$$

and can be rewritten as

$$\det \begin{bmatrix} P_0 & u_1 & \mathbf{0} \\ P_1 - P_0 & -u_0 & u_1 \end{bmatrix} = 0. \tag{2}$$

The first order continuous multilinear constraint is equivalent to

$$\det \begin{bmatrix} P & u & \mathbf{0} \\ P' & u' & u \end{bmatrix} = 0 \tag{3}$$

2.1 Vehicle and camera motion

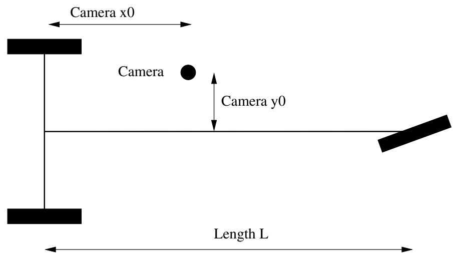


Fig. 1. Vehicle geometrical parameters

If odometry is to be used for navigation it is important to choose a vehicle for which the wheels are non-slipping. One such vehicle is a three-wheeled cart which is controlled by front wheel velocity v and angle s. For the motion of a camera centred in (x_0, y_0) relative to the centre of the rear axis we have [7]

$$\begin{bmatrix} v_x \\ v_y \\ \dot{\theta} \end{bmatrix} = v \begin{bmatrix} -x_0 \sin(s)/L \\ y_0 \sin(s)/L - \cos(s) \\ v \sin(s)/L \end{bmatrix} .$$

To simplify coming calculations we substitute $l = 1/L$, $X = x_0/L$, $Y = y_0/L$ and get

$$\begin{bmatrix} v_x \\ v_y \\ \dot{\theta} \end{bmatrix} = v \begin{bmatrix} -X \sin(s) \\ Y \sin(s) - \cos(s) \\ v \sin(s) l \end{bmatrix} .$$

As seen, all motion and motion derivatives can be computed from the signals (v, s) and the parameters (l, X, Y). With A the (3×3) camera calibration and orientation matrix relative to the vehicle coordinate system it is now possible to compute the (3×4) camera matrices P and $\dot{P}$. They are

$$P = A \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

and

$$\dot{P} = A \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & \dot{\theta} & -Xv\sin(s) \\ 0 & -\dot{\theta} & 0 & Yv\sin(s) - v\cos(s) \end{bmatrix} .$$

The equations for a system with differential steering are very similar.

3 Calibrating the Vehicle

It is now possible to assemble the continuous multilinear constraint

$$\det \underbrace{\begin{bmatrix} A & \mathbf{0} & u & \mathbf{0} \\ A\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \dot{\theta} \\ 0 & -\dot{\theta} & 0 \end{bmatrix} & A\begin{bmatrix} 0 \\ -Xv\sin(s) \\ Yv\sin(s) - v\cos(s) \end{bmatrix} & u^{(1)} & u \end{bmatrix}}_{M(t,x)} = 0. \tag{4}$$

The matrix $M(t,x)$ is polynomial in $x = (l, X, Y, , \{a_i\})$. Therefore $\det M(t,x)$ is polynomial in x. As equation (4) has to be fulfilled for all times we have one equation for every time and only 12 unknowns. In order to avoid the trivial solution we impose $||A||_F = 1$. To solve this problem in a least squares sense, i.e. to solve

$$\min_{\substack{x \\ ||A||_F = 1}} \underbrace{\sum_i (\det M(t_i, x))^2}_{F(x)}$$

we compute $F(x)$. Unfortunately $F(x)$ is of to high degree and contains to many coefficients to be easily optimised. For this reason a two stage method has been implemented assuming that the intrinsic parameters of the camera are known, that is, A is a rotation matrix.

Step 1, estimating one axis of the camera

Motion is restricted to pure translation by setting $s = 0$ and $v \neq 0$. With $P_0 = \begin{bmatrix} I \ \mathbf{0} \end{bmatrix}$ it is possible to write $P_1 = \begin{bmatrix} I \ T \end{bmatrix}$ where T is a translation in the forward direction of the vehicle and equation (2) can now be rewritten

$$\det \begin{bmatrix} -AT & -u_0 & u_1 \end{bmatrix} = 0. \tag{5}$$

The translation T is along the length-axis of the vehicle so $T = ||T|| \begin{bmatrix} 0 \ 0 \ 1 \end{bmatrix}$. The camera rotation

$$A = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 \end{bmatrix},$$

with $A_3 = \begin{bmatrix} A_3, a_6, a_9 \end{bmatrix}^T$ the third column of A. Equation (5) can be rewritten as

$$\det \begin{bmatrix} A_3 & u_0 & u_1 \end{bmatrix} = 0. \tag{6}$$

This equation has to be fulfilled for all pairs of measurements along a straight path. An optimal value for A_3 is found by minimising

$$\sum_{i,j} \det \begin{bmatrix} A_3 & u_i & u_j \end{bmatrix}^2. \tag{7}$$

Here A_3 is a column of a rotation matrix hence $||A_3|| = 1$. The fastest way to perform this minimisation is to compute the sum symbolically with A_3 as unknowns and then use svd [2] to find the minima.

Step 2, estimating the remaining camera parameters

Assuming internal calibration of the camera to be known and A_3 computed up to sign. As $A^T A = 1$ and $\det(A) = 1$ there is only one degree of freedom left for A. Introduce

$$B = \begin{bmatrix} \frac{-a_3 * a_9}{\alpha} & \frac{a_6}{\alpha} & a_3 \\ \frac{-a_6 * a_9}{\alpha} & \frac{-a_3}{\alpha} & a_6 \\ \alpha & 0 & a_9 \end{bmatrix}$$

with

$$\alpha = \sqrt{a_3^2 + a_6^2}$$

and

$$C(\phi) = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

All possible A can now be written

$$\begin{aligned} A(\phi) &= BC(\phi) \\ &= \begin{bmatrix} -\frac{a_3 a_9 \cos(\phi) + a_6 \sin(\phi)}{\alpha} & \frac{-a_3 a_9 \sin(\phi) + a_6 \cos(\phi)}{\alpha} & a_3 \\ \frac{-a_6 a_9 \cos(\phi) + a_3 \sin(\phi)}{\alpha} & -\frac{a_6 a_9 \sin(\phi) + a_3 \cos(\phi)}{\alpha} & a_6 \\ \alpha \cos(\phi) & \alpha \sin(\phi) & a_9 \end{bmatrix}. \end{aligned}$$

Using this A the continuous multilinear constraint i.e. $\det(M)$ in equation (4) is computed. It turns out that $\det(M)$ is a linear combination of 11 monomials, that is

$$\det(M) = \sum_{i=1}^{11} c_i q_i(\cos\phi, \sin\phi, l, X, Y).$$

When computing $\det(M)^2$ this is a linear combination of 44 monomials, that is,

$$(\det(M)^2) = \sum_{i=1}^{44} C_i p_i(\cos\phi, \sin\phi, l, X, Y).$$

The polynomial $(\det(M))^2$ is of total degree less than 8 and has the following properties

- The degree of trigonometric components is a most 4
- The degree of l is a most 2
- The degree of (X, Y) is a most 2

These coefficients can be added for several observations and used to calculate a least squares solution for our parameters given our observations.
The coefficients can be found in [7].

Optimization Step In the second step $F(l, X, Y)$ is a second order polynomial X and Y with computed from (l, ϕ) . The function F is written

$$F(l, X, Y) = \begin{bmatrix} X & Y \end{bmatrix} \alpha(l, \phi) \begin{bmatrix} X \\ Y \end{bmatrix} + \beta(l, \phi) \begin{bmatrix} X \\ Y \end{bmatrix} + \gamma(l, \phi)$$

minimizing this over (X, Y) and assuming $\det(\alpha) \neq 0$ we get

$$\min_{(X,Y)} F(l, \phi, X, Y) = -\frac{1}{4}\beta(l, \phi)'\alpha(l, \phi)^{-1}\beta(l, \phi) + \gamma(l, \phi)$$

which enables us to optimise in two variables instead of in four variables.

4 Numerical Experiments

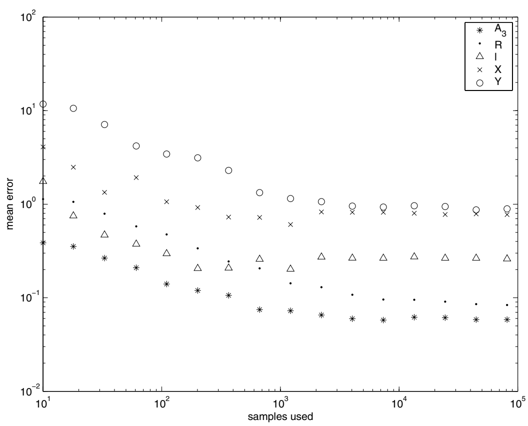


Fig. 2. Error reduction by increased sample size

Code used for the experiments below is available on the authors home-page [6].

Step 1

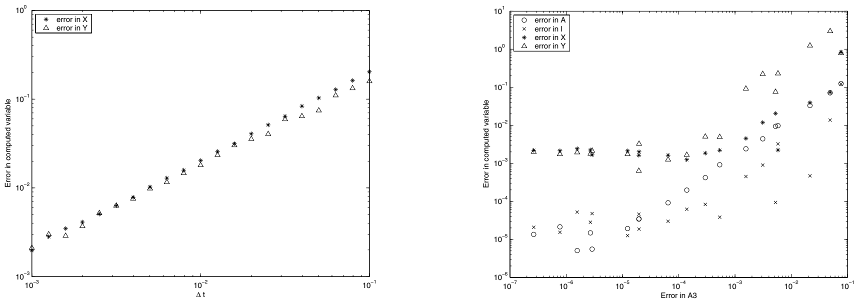
This step was tested with white noise in the data. It is clear from these experiments that the estimate for A_3 is improved by using a higher number of samples in the computation. This effect is visible for the stars of figure 2. The errors can also be reduced by computing the mean over several experiments.

Step 2

In this step there are problems coming both from errors in the first step, bias errors from using time-steps that are not infinitesimal and noise in image measurements that are amplified by the use of small time-steps. As in step 1 is is possible to reduce the error by increasing the sample size, this is shown in figure 2.

As seen in figure 3(a) there is a bias when determining (X, Y). This errors comes from that the equations used in step 2 assume time-steps to be infinitesimal. The reason for not using infinitesimal time-steps is that this would lead to an infinite amplification of signal noise. In the estimates of inverse length l this bias is very weak while it disappears when estimating camera matrix A.
In this theoretical setting the bias can be eliminated by Richardson extrapolation.

When there are errors in the computation of A_3 these errors propagate to the second step. In figure 3(b) is shown the effect of errors added to A_3 on the variables computed in the second step.



(a) Bias in X and Y against time-step used

(b) Effects of distortions in A_3

Fig. 3.

5 Summary and Conclusions

The method is a nice and new alternative to full reconstruction and works well at least for synthetic data. It is possible to use a very similar approach for a 4 wheeled cart with differential steering.

Further Work

It would be interesting to expand the system for cameras that are to a larger extent unknown. This should be quite straight forward but quite messy as the

number of polynomial coefficients will grow substantially. The elimination of the bias is important and it would be interesting if it was possible to explicitly calculate the error without use of the Richardson extrapolation scheme.
Of obvious interest is to test the method on a real AGV as only this can show if the results presented are relevant.

References

1. O. Faugeras and B. Mourrain. On the geometry and algebra of the point and line correspondences between n images. In *Proc. 5th Int. Conf. on Computer Vision, MIT, Boston, MA*, pages 951–956, 1995.
2. G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 1996.
3. A. Heyden and K. Åström. A canonical framework for sequences of images. In *Proc. IEEE workshop on Representation of Visual Scenes, MIT, Cambridge, MA, USA*, pages 45–52, 1995.
4. H. Malm and A. Heyden. Hand-eye calibration from image derivatives. In *Proc. 6th European Conf. on Computer Vision, Dublin, Ireland*, volume 2, pages 493–507, June 2000.
5. M. Oskarsson. One-dimensional retina vision with applications in structure and motion estimation. Licentiate thesis, Centre for Mathematical Sciences, Lund Institute of Technology, Sweden, 2000.
6. H. Stewenius. Homepage.
http://www.maths.lth.se/matematiklth/personal/stewe/.
7. H. Stewenius. Studies in mobile camera calibration. Licentiate thesis, Centre for Mathematical Sciences, Lund Institute of Technology, Sweden, 2003.
8. B. Triggs. Matching constraints and the joint image. In *Proc. ICCV'95*, pages 338–343. IEEE Computer Society Press, 1995.
9. R. Y. Tsai and R. K. Lenz. A new technique for fully autonomous and efficient 3d robotics hand/eye calibration. *IEEE Trans. Robotics and Automation*, 5(3):345–358, 1989.