HAL
open science

Plane-Based Calibration for Linear Cameras

Jamil Drareni, Peter Sturm, Sébastien Roy

To cite this version:

Jamil Drareni, Peter Sturm, Sébastien Roy. Plane-Based Calibration for Linear Cameras. OMNIVIS 2008 - 8th Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras, Rahul Swaminathan and Vincenzo Caglioti and Antonis Argyros, Oct 2008, Marseille, France. inria-00321895

HAL Id: inria-00321895

https://inria.hal.science/inria-00321895v1

Submitted on 16 Sep 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Author manuscript, published in "Dans The 8th Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras - OMNIVIS (2008)"

inria-00321895, version 1 - 16 Sep 2008

Plane-Based Calibration for Linear Cameras

Jamil Drareni, Peter Sturm, Sebastien Roy

INRIA, Rhone-Alpes, France

Abstract. Linear or 1D cameras are used in several areas such as industrial inspection and satellite imagery. Since 1D cameras consist of a linear sensor, a motion (usually perpendicular to the sensor orientation) is performed in order to acquire a full image. In this paper, we present a novel linear method to estimate the intrinsic and extrinsic parameters of a 1D camera using a planar object. As opposed to traditional calibration scheme based on 3D-2D correspondences of landmarks, our method uses homographies induced by the images of a planar object. The proposed algorithm is linear, simple and produces good results as shown by our experiments.

1 Introduction

2D perspective cameras are very common in computer vision thanks to their availability and versatility. For some applications such as industrial inspection, panorama acquisition and satellite imagery, linear (or pushbroom [1]) cameras are sometimes preferred over the conventional 2D cameras. This choice might be motivated by the need for a higher frame rate or a better resolution. If the acquired images are meant for a 3D euclidean reconstruction or metrology purposes, a camera calibration is necessary. As detailed in section 2 linear cameras have a specific model thus, standard 2D camera calibration methods can no longer be used to recover internal parameters.

Classical calibration methods use mappings of 3D feature points on a calibration rig and their projections on the image to infer the internal parameters of a camera [2, 3]. These methods are not very flexible because they use a specially designed calibration rig and often, features are manually selected.

In the last decade, new plane-based calibration methods have been introduced [4, 5]. They enjoyed a growing popularity in the computer vision community due to their stability and their higher ease of use. In fact, the calibration can be done with an off-the-shelf planar object and a printed checkboard.

Despite the several improvements that plane-based calibration methods went through [6–8], none of these works tackled the calibration of linear cameras. In fact, the predominant method for 1D camera calibration was proposed by Hartley et al. [9, 1] and supposes a mapping between 3D landmarks and their projections in the image. In this paper, we present a novel method to fully calibrate a 1D camera using a planar object. Our method is linear, fast and simple to implement. To our best knowledge, the presented plane-based 1D camera calibration is the first of its kind.

For the rest of the paper, the terms 1d camera, linear camera and pushbroom camera will be used equally.

The remaining of the paper is organized as follows, in section 2, the linear camera model is described. Section 3, presents the mathematical derivation and the algorithm of the plane-based calibration for linear cameras . The experiments and results are reported in section 4 and we finally summarize our conclusion in section 5.

2 Camera Model

Although the motivation behind our work was to calibrate a flatbed scanner using a pushbroom model, the presented algorithm along with the mathematical derivations still hold for any linear camera provided that the sensor undergoes a motion orthogonal to its orientation.

In general, a 1D camera consists of a linear array of sensors (such as CCD) recording an image projected by an optical system. A displacement of the sensor (usually orthogonal to the sensor) is required

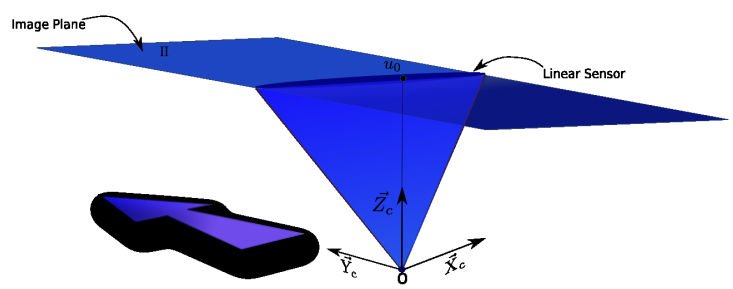

Fig. 1. A typical linear camera.

We make the same reasonable assumption as in [9] regarding the sensor motion. We suppose its velocity constant.

Without loss of generality, we set up the local camera coordinate system as depicted in the figure 1.

Let the point $(u, v, 1)^\mathsf{T}$ be the projection of the 3D point $(X, Y, Z)^\mathsf{T}$ in the camera image plane. The perspective projection of the coordinate u along the sensor can me modelled with a 2×3 projection matrix P:

$$\begin{pmatrix} u \\ 1 \end{pmatrix} \sim \underbrace{\begin{pmatrix} f & u_0 & 0 \\ 0 & 1 & 0 \end{pmatrix}}_{\mathsf{P}} \begin{pmatrix} X \\ Z \\ 1 \end{pmatrix} \tag{1}$$

inria-00321895, version 1 - 16 Sep 2008

inria-00321895, version 1 - 16 Sep 2008

The parameters f and u_0 are respectively the focal length and the optical center of the linear sensor.

As the sensor sweeps the scene, a 2D image is formed by stacking the 1D images obtained through the successive camera positions. Since the speed of the camera is assumed constant, the v coordinates is related to Y by a scaling factor s that depends on the speed of the sensor:

$$v = sY \tag{2}$$

If we combine (1) and (2) in a single matrix, the complete projection of a 3D point $(X, Y, Z)^\mathsf{T}$ is expressed as:

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} fX + u_0 Z \\ sYZ \\ Z \end{pmatrix} \sim \underbrace{\begin{pmatrix} f & 0 & u_0 \\ 0 & s & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\mathsf{K}} \begin{pmatrix} X \\ YZ \\ Z \end{pmatrix} \tag{3}$$

Where K represents the sought intrensic camera matrix.

We can see from the above equation that the perspective coordinate u depends solely on X and its depth Z, whereas v the orthographic coordinate is directly related to Y and the scaling factor s. One can also observe the non-linearity of the projection equation in the 3D coordinates due to the YZ term. This is not surprising, since the projection is non-central. This precludes the use of a pinhole-based camera calibration.

3 Calibration With a Planar Grid

Let us consider a point $(a, b, 0)^\mathsf{T}$ on the grid. It is mapped into the camera's coordinate system as $(X, Y, Z)^\mathsf{T}$ by a rigid transform:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \mathsf{R} \begin{pmatrix} a \\ b \\ 0 \end{pmatrix} + \mathbf{t} \tag{4}$$

Where R is a 3×3 rotation matrix and $\mathbf{t}$ a translation vector. Notice that, since the considered point lies on the grid, its third coordinate is null. Hence, the entries of the third column of R are zeroed and the equation (4) in homogeneous coordinates simplifies as:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} \mathbf{R}_1 & \mathbf{R}_2 & \mathbf{t} \end{pmatrix} \begin{pmatrix} a \\ b \\ 1 \end{pmatrix} = \begin{pmatrix} ar_{11} + br_{12} + t_1 \\ ar_{21} + br_{22} + t_2 \\ ar_{31} + br_{32} + t_3 \end{pmatrix} \tag{5}$$

Where $\mathbf{R}_1$ and $\mathbf{R}_2$ are the two first columns of R.

As stated before, the non-central nature of the camera makes it impossible to establish a linear mapping between points on the grid and their images on the camera plane. For instance, $(u, v, 1)^\mathsf{T}$ is expressed from equations (3) and (5) as:

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \sim \mathsf{K} \begin{pmatrix} X \\ YZ \\ Z \end{pmatrix} \tag{6}$$
$$= \mathsf{K} \begin{pmatrix} ar_{11} + br_{12} + t_1 \\ a(r_{21}t_3 + r_{31}t_2) + b(r_{22}t_3 + r_{32}t_2) + t_2t_3 + a^2r_{21}r_{31} + b^2r_{22}r_{32} + ab(r_{21}r_{32} + r_{22}r_{31}) \\ ar_{31} + br_{32} + t_3 \end{pmatrix}$$

A little trick to circumvent this problem is to express the points in a higher dimensional space via the so-called "lifted" coordinates. In our case, the point $(a, b, 1)^\mathsf{T}$ "lifts" (according to their Veronese mapping) to $(a, b, 1, a^2, b^2, ab)^\mathsf{T}$. Thus, the equation (6) becomes:

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \sim \mathsf{K} \cdot \underbrace{\begin{pmatrix} r_{11} & r_{12} & t_1 & 0 & 0 & 0 \\ r_{21}t_3 + r_{31}t_2 & r_{22}t_3 + r_{32}t_2 & t_2t_3 & r_{21}r_{31} & r_{22}r_{32} & r_{21}r_{32} + r_{22}r_{31} \\ r_{31} & r_{32} & t_3 & 0 & 0 & 0 \end{pmatrix}}_{\mathsf{T}} \begin{pmatrix} a \\ b \\ 1 \\ a^2 \\ b^2 \\ ab \end{pmatrix} \tag{7}$$

Which represents the complete projection equation of a point on the grid expressed in its lifted coordinates.

The homography $\mathsf{H} \sim \mathsf{KT}$ that maps point on the grid and its image has 6 zeroed entries. The remaining 12 non-zero entries can be estimated up to a scale factor using 6 or more point matches as explained in the next subsection.

3.1 Estimate the Homography

We recall from equation 7 that the mapping between grid points and image points is represented by the homography H as:

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \sim \mathsf{H} \cdot \begin{pmatrix} a \\ b \\ 1 \\ a^2 \\ b^2 \\ ab \end{pmatrix} \tag{8}$$

If we multiply both hands of the above equation by $\left[\begin{pmatrix} u \\ v \\ 1 \end{pmatrix}\right]_\times$, the cross product skew matrix, we get a homogeneous equation system that upon simplifications yields the following linear and homogeneous equation system in the entries of H:

$$\begin{pmatrix} 0 & 0 & 0 & a & b & 1 & a^2 & b^2 & ab & -av & -bv & -v \\ a & b & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -au & -bu & -u \\ -av & -bv & -v & au & bu & u & a^2u & b^2 & abu & 0 & 0 & 0 \end{pmatrix} \mathbf{h} = \mathbf{0} \tag{9}$$

Where $\mathbf{h}^\mathsf{T} = (\mathbf{h}_{11}, \mathbf{h}_{12}, \mathbf{h}_{13}, \mathbf{h}_{21}, \mathbf{h}_{22}, \mathbf{h}_{23}, \mathbf{h}_{24}, \mathbf{h}_{25}, \mathbf{h}_{26}, \mathbf{h}_{31}, \mathbf{h}_{32}, \mathbf{h}_{33})$ is the vector that contains the non-zero entries of H.

inria-00321895, version 1 - 16 Sep 2008

It is easy to see that only two equations are linearly independent. For instance, the third row can be obtained by adding the first and the second row, scaled respectively by u and v. Thus, given at least 6 matches between grid points and their images, H can be solved using 2 equations from the system (9) per match.

3.2 Extracting the Principal Point and the Focal Length

We shall now show how the camera's internal parameters are extracted from the homographies computed in the previous subsection. Let us recall the explicit form of the homography H:

$$\mathsf{H} = \lambda \begin{pmatrix} fr_{11} + u_0 r_{31} & fr_{12} + u_0 r_{32} & ft_1 + u_0 t_3 & 0 & 0 & 0 \\ s(r_{21}t_3 + r_{31}t_2) & s(r_{22}t_3 + r_{32}t_2) & st_2 t_3 & sr_{21}r_{31} & sr_{22}r_{32} & s(r_{21}r_{32} + r_{22}r_{31}) \\ r_{31} & r_{32} & t_3 & 0 & 0 & 0 \end{pmatrix}$$

The scalar λ is added because the homography H can only be retrieved up to a scale factor.

One can notice that $\bar{\mathsf{R}}$, the two first rotation's columns can be expressed as:

$$\bar{\mathsf{R}} = \begin{pmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \\ r_{31} & r_{32} \end{pmatrix} = \begin{pmatrix} \frac{h_{11} - u_0 h_{31}}{\lambda f} & \frac{h_{12} - u_0 h_{32}}{\lambda f} \\ \frac{h_{24}}{s h_{31}} & \frac{h_{25}}{s h_{32}} \\ \frac{h_{31}}{\lambda} & \frac{h_{32}}{\lambda} \end{pmatrix}$$

From the above equation, $\bar{\mathsf{R}}$ can be expressed as a product of two matrices (up to a scale factor) L that depends on internal parameters and M:

$$\mathsf{L} = \begin{pmatrix} s & 0 & -su_0 \\ 0 & \lambda f & 0 \\ 0 & 0 & sf \end{pmatrix} \mathsf{M} = \begin{pmatrix} H_{11} & H_{12} \\ \frac{H_{24}}{H_{31}} & \frac{H_{25}}{H_{32}} \\ H_{31} & H_{32} \end{pmatrix}$$

The product of $\bar{\mathsf{R}}$ with its transpose is a 2×2 identity matrix due to the orthogonality of its columns. Thus, we have:

$$\bar{\mathsf{R}}^{\mathsf{T}} \bar{\mathsf{R}} = \mathtt{I}_{2\times 2} \sim \mathsf{M}^{\mathsf{T}} \mathsf{L}^{\mathsf{T}} \mathsf{L} \mathsf{M} \tag{10}$$

The matrix L is related to the above calibration matrix K, with the notable fact that it also includes the scalar λ. Note that λ will be different for each view, as opposed to the 3 intrinsic parameters f, s and u_0 which remain the same.

Let us define the matrix X as:

$$\mathsf{X} = \mathsf{L}^{\mathsf{T}} \mathsf{L} = \begin{pmatrix} s^2 & 0 & -s^2 u_0 \\ 0 & \lambda^2 f^2 & 0 \\ -s^2 u_0 & 0 & s^2(u_0^2 + f^2) \end{pmatrix}$$

For ease of notation, we introduce the intermediate variables v_1, v_2, v_3, v_4 as:

$$\begin{aligned} v_1 &= s^2 \\ v_2 &= -s^2 u_0 \\ v_3 &= s^2(u_0^2 + f^2) \\ v_4 &= \lambda^2 f^2 \end{aligned}$$

inria-00321895, version 1 - 16 Sep 2008

The equation (10) gives 2 constraints on X that can be written as:

$$\left.\begin{aligned}\left(\mathsf{M}^{\mathsf{T}}\mathsf{X}\mathsf{M}\right)_{12} &= 0\\ \left(\mathsf{M}^{\mathsf{T}}\mathsf{X}\mathsf{M}\right)_{11} - \left(\mathsf{M}^{\mathsf{T}}\mathsf{X}\mathsf{M}\right)_{22} &= 0\end{aligned}\right\} \tag{11}$$

Which in turn can be expressed in terms of the intermediate variables v_i as:

$$\begin{pmatrix} m_{11}m_{12} & m_{11}m_{32}+m_{12}m_{31} & m_{31}m_{32} & m_{21}m_{22} \\ m_{11}^2-m_{12}^2 & 2(m_{11}m_{31}-m_{12}m_{32}) & m_{31}^2-m_{32}^2 & m_{21}^2-m_{22}^2 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{12}$$

With at least two different views of a grid, the v_i can be computed up to a scaling factor. Bare in mind that v_4 is different at each view because of the homography scaling factor λ. Once the v_i computed, the principal point and the focal length are simply computed as:

$$u_0 = -\frac{v_2}{v_1} \tag{13}$$

$$f = \sqrt{\frac{v_3}{v_1} - u_0^2} = \sqrt{\frac{v_3}{v_1} - \frac{v_2^2}{v_1^2}} = \sqrt{\frac{v_1 v_3 - v_2^2}{v_1^2}} \tag{14}$$

3.3 Extracting the Scaling factor and the Extrinsic Parameters

Now that we have extracted the focal length and the principal point, we will show how the scaling factor s along with the extrinsic parameters (rotation matrix R and translation vector $\mathbf{t}$) can be computed using more constraints.

Let us define the matrix A_i as:

$$\mathsf{A}_i = \lambda_i \begin{pmatrix} r_{11} & r_{12} & t_1 & 0 & 0 & 0 \\ s(r_{21}t_3 + r_{31}t_2) & s(r_{22}t_3 + r_{32}t_2) & st_2t_3 & sr_{21}r_{31} & sr_{22}r_{32} & s(r_{21}r_{32} + r_{22}r_{31}) \\ r_{31} & r_{32} & t_3 & 0 & 0 & 0 \end{pmatrix} \tag{15}$$

The subscript i refers to the ith view of the calibration grid. We can first notice that:

$$\begin{aligned} t_{1i} &= \frac{a_{13}}{\lambda_i} \\ t_{2i} &= \frac{a_{23}}{sa_{33}} \\ t_{3i} &= \frac{a_{33}}{\lambda_i} \end{aligned} \tag{16}$$

It's easy to see that $\bar{\mathsf{R}}_i$ (the two first columns of R_i as defined in the previous subsection) can be expressed as:

inria-00321895, version 1 - 16 Sep 2008

$$\bar{\mathsf{R}}_i = \begin{pmatrix} \frac{1}{\lambda_i} & & \\ & \frac{1}{sa_{33i}} & \\ & & \frac{1}{\lambda_i} \end{pmatrix} \begin{pmatrix} a_{11i} & a_{12i} \\ a_{21i} - a_{31i}\frac{a_{23i}}{a_{33i}} & a_{22i} - a_{32i}\frac{a_{23i}}{a_{33i}} \\ a_{31i} & a_{32i} \end{pmatrix} \tag{17}$$

$$= \begin{pmatrix} \frac{1}{\lambda_i} & & \\ & \frac{1}{s} & \\ & & \frac{1}{\lambda_i} \end{pmatrix} \underbrace{\begin{pmatrix} a_{11i} & a_{12i} \\ \frac{a_{21i}a_{33i} - a_{31i}a_{23i}}{a_{33i}^2} & \frac{a_{22i}a_{33i} - a_{32i}a_{23i}}{a_{33i}^2} \\ a_{31i} & a_{32i} \end{pmatrix}}_{\mathsf{B}_i} \tag{18}$$

$$\tag{19}$$

Where a_{xyi} are the elements of the matrix A_i.

As in the previous subsection, we once again make use of the orthogonality of the rotation matrix $\bar{\mathsf{R}}_i$ to gain constraints on λ_i and $\frac{1}{st_{3i}}$. For instance, one notices that:

$$\bar{\mathsf{R}}_i^{\mathsf{T}} \bar{\mathsf{R}}_i = \mathsf{B}_i{}^{\mathsf{T}} \begin{pmatrix} \frac{1}{\lambda_i^2} & & \\ & \frac{1}{s^2} & \\ & & \frac{1}{\lambda_i^2} \end{pmatrix} \mathsf{B}_i = \mathtt{I}_{2\times 2}$$

The above result gives 3 linear equations in $\frac{1}{\lambda_i^2}$ and $\frac{1}{s^2}$. Since we solved for $\frac{1}{\lambda_i^2}$, the scaling factor λ_i is extracted up to a sign.

So far, only the 3 first columns of A_i have been used. In order to extract the real λ_i from the 2 possible solutions, the 3 last columns of A_i are going to be used. We proceed with the following simple steps for each possible solution:

- Compute $\bar{\mathsf{R}}_i$ (cf. equation (19)) and A_i.
- Compute t_{1i} and t_{3i} as defined in (17).
- Compute the following term:

$$\Delta = (a_{24i} - sr_{21i}r_{31i})^2 + (a_{25i} - sr_{22i}r_{32i})^2 + (a_{26i} - s(r_{21i}r_{32i} + r_{22i}r_{31i}))^2$$

The real solution is the one that leads to the smallest Δ. By the definition of a_{24i}, a_{25i} and a_{26i} (see equation (15)) and in an ideal noiseless case, Δ vanishes. The last but not the least, once a couple (s, λ_i) computed, one can notice that $(-s, -\lambda_i)$ minimizes also the term Δ. This later ambiguity corresponds to the mirror-pose solution. Given our choice of coordinate system, visible points must have positive Z-coordinate, thus we pick the solution that gives a positive t_{3i}.

Finally, t_{2i} is computed from 17 and the third column of the rotation matrix is obtained by a simple cross-product of the two columns of $\bar{\mathsf{R}}_i$. The orthonormality of the final rotation matrix R_i can be enforced by SVD.

Notice that, as opposed to the reference calibration method [9], the proposed method estimates the scaling factor s related to the speed of the linear sensor.

inria-00321895, version 1 - 16 Sep 2008

3.4 Complete Plane-Based Calibration Algorithm

In this subsection we present the complete plane-based algorithm for linear cameras calibration. From n view of a calibration grid:

1. Estimate the projection matrices H_i for all n views, using point matches and the relation
$$\begin{pmatrix} u_{ij} \\ v_{ij} \\ 1 \end{pmatrix} \sim \mathsf{H}_i \begin{pmatrix} a_j \\ b_j \\ 1 \\ a_j^2 \\ b_j^2 \\ a_j b_j \end{pmatrix}$$
where j is an index for calibration points. The estimation of H_i is equivalent to the so-called DLT (Direct Linear Transform) and can be done by solving a linear equation system.
2. Compute matrices M_i according to equation (12).
3. Form the following linear equation system of dimension $2n \times (3+n)$:
$$\begin{pmatrix} m_{1,11}m_{1,12} & m_{1,11}m_{1,32} + m_{1,12}m_{1,31} & m_{1,31}m_{1,32} & m_{1,21}m_{1,22} & & \\ \vdots & \vdots & \vdots & & \ddots & \\ m_{n,11}m_{n,12} & m_{n,11}m_{n,32} + m_{n,12}m_{n,31} & m_{n,31}m_{n,32} & & & m_{n,21}m_{n,22} \\ M_{1,11}^2 - M_{1,12}^2 & 2(m_{1,11}m_{1,31} - m_{1,12}m_{1,32}) & M_{1,31}^2 - M_{1,32}^2 & M_{1,21}^2 - M_{1,22}^2 & & \\ \vdots & \vdots & \vdots & & \ddots & \\ M_{n,11}^2 - M_{n,12}^2 & 2(m_{n,11}m_{n,31} - m_{n,12}m_{n,32}) & M_{n,31}^2 - M_{n,32}^2 & & & M_{n,21}^2 - M_{n,22}^2 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ v_{4,1} \\ v_{4,2} \\ \vdots \\ v_{4,n} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}$$
and solve it to least squares.
4. From the v_i, extract the intrinsic parameters f and u_0 according to (13) and (14).
5. Compute s and the extrinsic parameters according to the algorithm of subsection 3.3.
6. Optional but recommended: non-linear optimization of all unknowns, i.e. intrinsic and extrinsic parameters, by minimizing the reprojection errors.

4 Experimental Results

The proposed algorithm has been tested on both synthetic data and real data. Both tests are detailed in the next two subsections.

4.1 Computer Simulations

We performed several tests of our algorithm using synthetic data. Throughout all the experiments, we used a planar calibration grid of $10 \times 10 = 100$ corners. The image resolution, the focal length and the optical center of the virtual camera are 1000×1000, 1000 and 500 respectively.

We refer to the *"calibration volume"*, the bounding box that encloses the calibration grid's motions. Actually the most relevant parameter is not the bounding

inria-00321895, version 1 - 16 Sep 2008

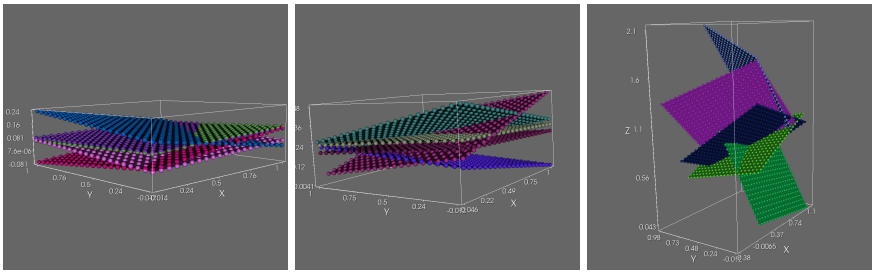

Fig. 2. An example of 3 calibration volume with increasing height. From left to right, 25%, 50% and 200% of the calibration length.

box volume itself but it's height. In our experiments, the volume height is expressed as a percentage of the grid's length. Some configuration examples with several calibration volume are depicted in Fig.2.

Sensitivity to noise level For this test, we used 10 planes oriented randomly in a calibration volume of 100% the size of the calibration grid. After projection, a gaussian noise with mean 0 and increasing standard deviation was added to the image points. The standard deviation σ varied from 0.2 to 2. As in [5], we performed 100 independent runs for each noise level and computed the average errors for both the focal length and the principale point. As we can see from Fig.3 the error increases almost linearly for both the focal and the optical center, u_0. For an noise level of $\sigma = 0.5$ the errors in the focal and the optical center is less than 4 pixels which represents (given our camera characteristics) less than 0.8%.

Sensitivity to the number of planes In this test, the sensitivity of our method w.r.t the number of planes is investigated. We set the calibration volume height to 100% of the grid's length and we varied the number of planes from 2 to 20. The average errors (from 100 independent runs) for both the focal length and the optical center were estimated and reported on Fig.4 for a noise level of $\sigma = 0.5$ and $\sigma = 1.0$. We notice that the errors decrease when more planes are used.

Sensitivity w.r.t the reconstruction volume In this last synthetic experiment we analyse the performance of our method with respect to the calibration volume, or more precisely the volume's height. For this test we used 10 calibration grids oriented randomly and varied the calibration volume height from 20% to 160% of the grid's length. This test was performed with a noise level of $\sigma = 0.5$ and $\sigma = 1.0$ (which is larger than the normal noise in practical calibration [5]). We can see from Fig.5 and Fig.6 that the volume's height affects the quality of the calibration. In fact the errors decrease when a higher reconstruction volume is used. This is primarily due to the fact that a higher reconstruction volume permits a higher motion degree which guarantees a better sampling of the rotation space.

inria-00321895, version 1 - 16 Sep 2008

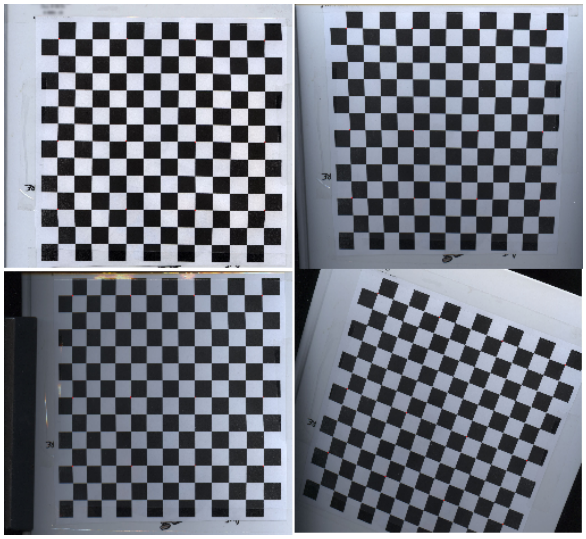

Fig. 3. Error w.r.t the noise level in the image point.

4.2 Real Data

We tested the proposed algorithm on an Epson V200 flatbed scanner. The manufacturer claims that the scanner is suited for scanning 3D objects thanks to its depth of field and adapted optic. We thus, modeled the scanner as a push-broom camera and used the proposed algorithm to retrieve its intrinsic parameters using a planar grid. Some of the acquired images are depicted on figure Fig. 7. The scans were done at a resolution of 300dpi (dot per inch), the grid's squares were half inch long each. Resulting images had a resolution of 2538x2328. Homographies were estimated by first detecting grid's features using OpenCV routines. To ensure a better numerical stability, points were normalized as suggested in [10]. We also calibrated the same scanner using the DLT method proposed by Hartley [9] [1]. In the later case, we scanned a 3D calibration rig and features were manually selected. Results and comparisons are reported in Table. 1

Table 1. Results of the flatbed scanner calibration using our method and the DLT method (see text).

Parameter	**DLT**	**Plane-Based**
Focal Length	2673.4	2659.7
Optical Center	1315.2	1299.5
Scale Factor	—	146.48

inria-00321895, version 1 - 16 Sep 2008

inria-00321895, version 1 - 16 Sep 2008

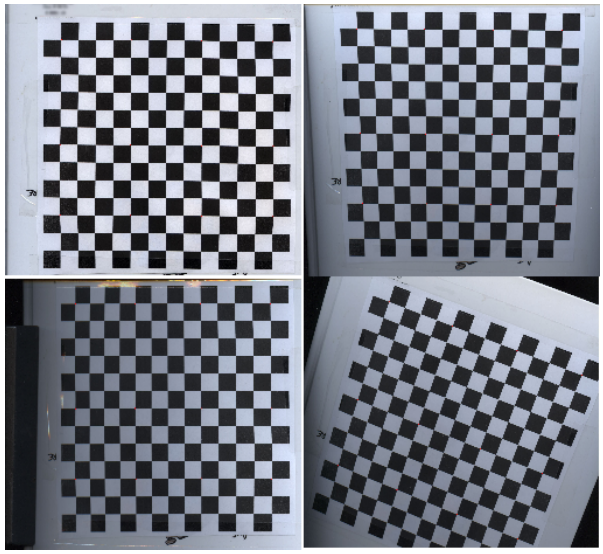

Fig. 4. Errors vs. the number of planes used ($\sigma = 0.5$).

Since no ground truth was available, we took as a reference the classical calibration method proposed by Hartley et al.[9, 1] and we can see that the focal length and the principle point estimated by our method are very close to the estimation made by Hartley's method (both parameters differ by less than 1.5%). Further, each square of the calibration grid measured 0.5 inch length and giving the fact that the tests were made at a resolution of 300dpi, the scaling factor s should be $s = 300 \times 0.5 = 150$ wich differs by only 2.4% from the scaling factor computed using our method.

5 Conclusion

In this paper we have presented a simple algorithm to calibrate a linear camera. The calibration is done using images of a planar grid acquired under different orientations. As opposed to the reference method, the proposed one estimates all thee internal parameters including the scaling factor.

References

1. Gupta, R., Hartley, R.I.: Camera estimation for orbiting pushbrooms. In: The Proc. of Second Asian Conference on Computer Vision. Volume 3. (1995)
2. Tsai, R.: An efficient and accurate camera calibration technique for 3-d machine vision. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition. (1986) 364–374
3. O. Faugeras, G.T.: Camera calibration for 3d computer vision. In: Int. Workshop on Machine Vision and Machine Intelligence. (1987) 240–247

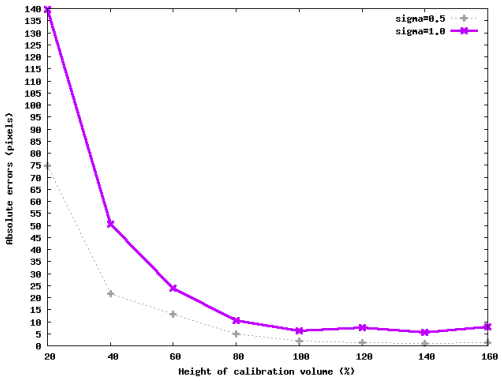

Fig. 5. Errors in focal length vs. the height of calibration volume.

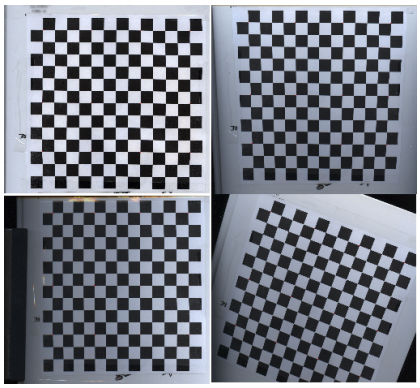

Fig. 6. Errors principle point vs. the height of calibration volume.

inria-00321895, version 1 - 16 Sep 2008

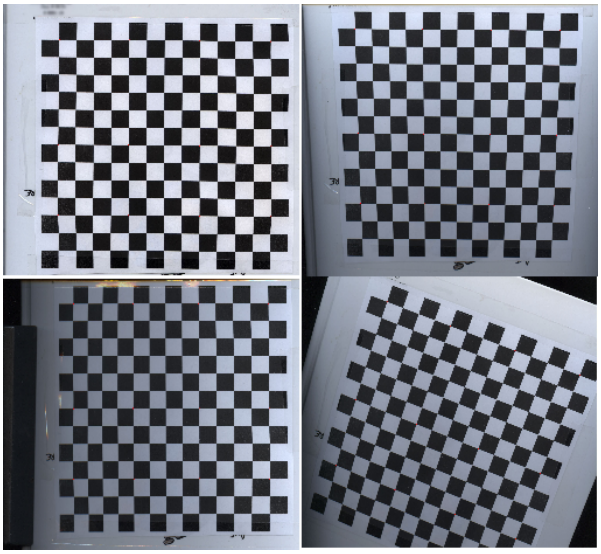

Fig. 7. Some images of a calibration grid acquired with a flatbed scanner (2538x2328)

4. Sturm, P., Maybank, S.: On plane-based camera calibration: A general algorithm, singularities, applications. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition. (1999) I: 432–437
5. Zhang, Z.: Flexible camera calibration by viewing a plane from unknown orientations. In: International Conference on Computer Vision. (1999) 666–673
6. Gurdjos, P., Payrissat, R.: Calibration of a moving camera using a planar pattern: A centre line-based approach for varying focal length. In: British Machine Vision Conference. (2001) Session 7: Geometry &. Structure
7. Wang, J., Shi, F., Zhang, J., Liu, Y.: Camera calibration from a single frame of planar pattern. In: Advanced Concepts for Intelligent Vision Systems. (2006) 576–587
8. Ricolfe Viala, C., Sanchez Salmeron, A.: Improving accuracy and confidence interval of camera parameters estimated with a planar pattern. In: International Conference on Image Processing. (2005) II: 1142–1145
9. Rajiv Gupta, R.I.H.: Linear pushbroom cameras. IEEE Transactions on Pattern Analysis and Machine Intelligence **19** (1997) 963–975
10. Hartley, R.: In defense of the eight-point algorithm. IEEE Transactions on Pattern Analysis and Machine Intelligence **19** (1997) 580–593

inria-00321895, version 1 - 16 Sep 2008