09-23

15th IAPR International Conference on Machine Vision Applications (MVA)
Nagoya University, Nagoya, Japan, May 8-12, 2017.

Low Cost Calibration of Stereo Line Scan Camera Systems

Erik Lilienblum, Sebastian Handrich, Ayoub Al-Hamadi
Otto-von-Guericke University
Magdeburg, Germany
{erik.lilienblum, sebastian.handrich, ayoub.al-hamadi}@ovgu.de

Abstract

We present a new calibration method for 3d measurement systems consisting of two co-planar aligned line scan cameras. As appliance we use a calibration target whose geometric 2d and 3d parameters are only approximately known. As our input data we take from that target a sufficient number of single captured image lines in different geometric positions. Then we calculate 2d points within the joint viewing plane of the cameras both by triangulation of corresponding image points and by intersection with the lines on the target surface. The distances between the triangulation points and the related intersection points are minimized by linearizing and least squares adjustment. In the result we obtain all relevant parameters of the inner and outer orientation as well as the 2d and 3d geometry parameters of the calibration target with high accuracy.

1 Introduction

Optical methods in industrial applications are quite often based on line scan cameras. Obvious benefits compared to matrix cameras are continuous processing, short integration times, high resolution and high reliability. This is true for both 2d and 3d applications. But while 2d applications are very common, 3d applications are rare. One reason is that most 3d applications need a very precise geometric calibration. In general, this is sophisticated for line scan systems.

While for matrix cameras there are a lot of well known calibration approaches [8] and even public implementations (for example OpenCV, MATLAB [1]) the calibration of line scan cameras is still problematic. The few known approaches either provide only a low accuracy or need high technical equipment such as costly produced and measured calibration targets or special movement devices. For example, in [3] approaches using linear movement are described. However, each divergence from the linearity may lead to an inaccuracy of the calibration. And for some applications the use of an appropriate linear unit is generally impossible. Other approaches like described in [7, 6] disclaim the movement at all. But they presume a well known calibration target with high precision both in 2d and in 3d. We can archieve the required precision only by measuring the target by a second system which is calibrated at least with the same accuracy. And normally, this is costly and one need considerable experience to avoid measuring faults.

The aim of the calibration method which we present in this paper is to disclaim both a known movement and a precisely measured target. This may be an important contribution to make 3d line-scan approaches (for example [2, 4, 5]) more applicable in machine vision.

2 Presumptions

2.1 Calibration Target

The design of our calibration target is based on the approach given in [6]. They describe an algorithm to calculate 3d points by intersection between straight lines of a pattern and the viewing plane of a line-scan camera. As suggested in this approach we build for experimental evaluations a target with four planes in two different heights with three coded line markers on each plane. The line markers are printed with a laser standard printer on white paper and laminated on the planes by hand. Figure 1 shows our calibration target, which is quite similar to [6] but not symmetric.

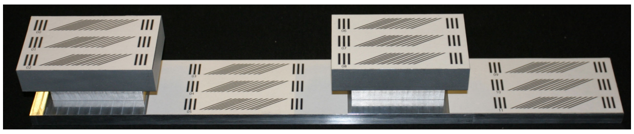

Figure 1. Calibration target

Of course, the geometric design of the target can be changed. Width, spacing and alignment of straight lines can be optimized for the system that we want to calibrate. The only presumption concerning the target is that we rely on an unambiguous mapping from 1d points on single captured image lines to 3d intersection points on the target surface. To avoid measuring the 2d and 3d geometry parameters of our self made target by a second system, we use only approximations. Thus, at the beginning of calibration process the accuracy of the 3d mapping is not very important.

2.2 System Setup and Target Views

In contrast to [6], our calibration approach works only on stereo systems with co-planar aligned cameras. The alignment should to be good but need not to be perfect. Additionally, the cameras have to be triggered so that both cameras of the stereo system capture their lines at the same time. There are no other presumptions for the system setup because we do not assume any defined, e.g. linear, movement.

As input data for our calibration we take a sufficient number of independent single captured image lines from the calibration target in different positions. We call it target views. Of course, the more calibration parameters we try to calculate the more target views we have to capture. But due to the independency each target view has its own outer orientation. Thus, we get a lot of additional parameters. The resulting relations are very complex and we need a systematic strategy, for our calibration algorithm to converge. Our suggestion for that strategy is drafted in figure 2.

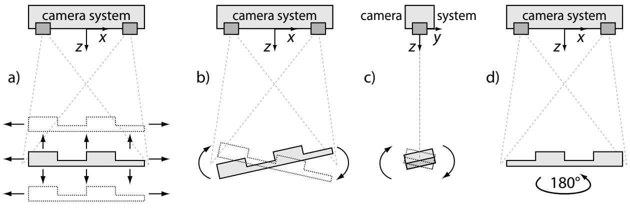

Figure 2. Different target views

We put the target in three different heights and in each height we shift the target to the left and to the right (2.a). Then we tilt the target around the y-axis (2.b) and around the x-axis (2.c) in two directions respectively. Thus, we get a lot of different positions which we double by turning the target for 180 degree around the z-axes (2.d). Additionally, we may capture in each position as many lines as we want on different y-coordinates. Of course, this provides a large amount of input data. But we need this amount to handle the approximation of the target parameters at the beginning and to get the desired accuracy at the end.

2.3 Camera Model

Theoretically, each line scan camera can be considered as a special case of a matrix camera because they differ only on the shape of the sensor while optics and geometry are the same. Thus, we can describe line scan cameras with the same camera model. As common in computer vision we use a pinhole model with radial symmetric distortion.

Normally, for each camera we need 6 external parameters (the rotation angles ω, φ, κ and the translation defined by the perspective center $\mathbf{O} = (O_x, O_y, O_z)^T$) and 5 internal parameters (the principle point (H_u, H_v), the camera constant c and two distortion parameters A', A''). Thus, for our stereo system consisting of two cameras we need 24 parameters. But this yields only for one single captured image line. For each further image line we need another 12 external parameters (6 for each camera).

However, we can reduce the number of free parameters if we take into account the co-planarity and the fix geometry of our stereo system. As shown in figure 3 we consider all relationships between the cameras in a sensor coordinate system whose xz-plane matches the joint viewing plane of the cameras.

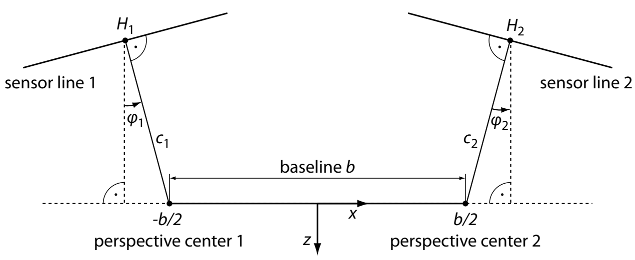

Figure 3. Internal parameter of the stereo system

To define joint internal and external sensor parameters we align the x-axis of our sensor coordinate system to the line between the perspective center of both cameras and fix the origin in the center. Thus, the only parameter for translation is the baseline b. Because we are located within the viewing plane the rotation needs only a single angle φ_i for each camera $i = 1, 2$. Of course, there is also no vertical principal point. All remaining parameters such as the horizontal principal point H_i, the camera constant c_i and the parameters for radial symmetric distortion A'_i, A''_i are directly taken from the separate camera models. The outer orientation relates to the stereo sensor as a whole and can be described by only 6 external parameters.

Naturally, our 2d stereo model is inaccurate, because in the real world the co-planarity cannot be assumed to be perfect. Thus we model additionally a symmetric deviation along the xz-plane with

$$\varepsilon(x, z) = a_x x + a_z z + a_0. \tag{1}$$

We neglect other inaccuracies like vertical distortions due to their very little influence on line scan cameras.

3 Optimization Approach

3.1 Parameter Sets

The presumptions in section 2 lead to different classes of parameters. The first parameter class describes the outer orientation. Let n_v be the number of different target views. Then we yield a set

$$\mathbf{p}_1 = \bigcup_{i=1}^{n_\mathrm{v}} \{\omega_i, \varphi_i, \kappa_i, x_i, y_i, z_i\} \tag{2}$$

of outer parameters, which defines for each target view the coordinate transformation from the 2d and 3d geometric parameters of the target into the sensor coordinate system shown in figure 3.

The second parameter class describes the internal parameters of the stereo system with

$$\begin{aligned} \mathbf{p}_2 = \{\, & b, c_1, c_2, \varphi_1, \varphi_2, h_1, h_2, \\ & A'_1, A'_2, A''_1, A''_2, a_x, a_y, a_0 \,\}. \end{aligned} \tag{3}$$

The calibration target is described by two parameter classes defining the 3d geometry of the planes and the 2d position of the lines on the plane surfaces separately. Because we use a self-made target, neither the planes nor the lines are perfect. Thus, we have to modulate curved surfaces and curved lines, respectively. To get a sufficient approximation we model each plane through a 3rd degree Bezier segment using 16 nodes in a fixed x-y-grid where only z-components are free. If we have n_p different planes of the target, then we yield a set of plane parameters with

$$\mathbf{p}_3 = \bigcup_{i=1}^{n_\mathrm{p}} \{z_i^{(1)}, z_i^{(2)}, \ldots, z_i^{(16)}\}. \tag{4}$$

For modulating the lines we take polynomials of 3rd degree. Because all lines are either vertical or diagonal we use for each line a function of the form

$$x = f(y) = ay^3 + by^2 + cy + d. \tag{5}$$

Let n_l be the number of all lines. Then we yield the set of line parameters with

$$\mathbf{p}_4 = \bigcup_{i=1}^{n_\mathrm{l}} \{a_i, b_i, c_i, d_i\}. \tag{6}$$

But the parameter sets $\mathbf{p}_3$ and $\mathbf{p}_4$ cannot describe the calibration target completely. In addition, we need a mapping

$$\pi(i) = j \tag{7}$$

which assigns to each line $i \le n_\mathrm{l}$ one plane $j \le n_\mathrm{p}$.

3.2 Objective Function

In computer vision many calibration approaches for single cameras use space resection calculating pixel residues for the numerical optimization of the inner and outer camera parameter. To increase the numeric stability we optimize in our approach the stereo system as a whole. Thus, it is more suitable to calculate residues using the distances between reconstructed points. Our residues are based on the distances between intersection points and triangulated points. An intersection point arises where a target line meets the joint viewing plane. We obtain the appropriate triangulation point by triangulation of corresponding pixels of the same target line. A schematic illustration of reconstructed points on the joint viewing plane is given in figure 4.

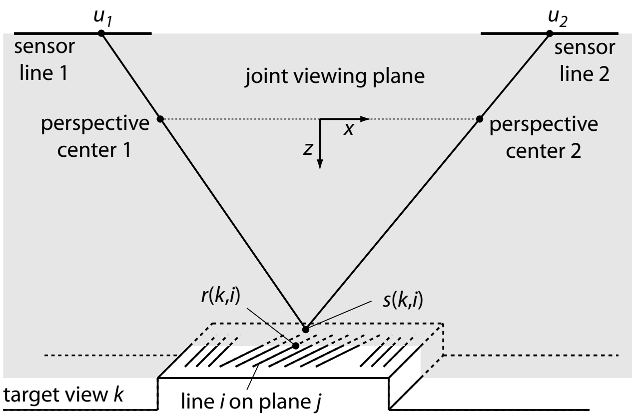

Figure 4. Intersection and triangulation points

Let $u_1(k,i)$ and $u_2(k,i)$ be corresponding pixels seeing in target view $k \le n_\mathrm{v}$ the same line $i \le n_\mathrm{l}$. Then we calculate the intersection point by

$$\begin{aligned} r(k,i) = f_{\mathrm{sec}}(&\omega_k, \varphi_k, \kappa_k, x_k, y_k, z_k, \\ &z^{(1)}_{\pi(i)}, \ldots, z^{(16)}_{\pi(i)}, a_i, b_i, c_i, d_i), \end{aligned} \tag{8}$$

which is a function working on standard geometry. And we calculate the triangulation point

$$\begin{aligned} s(k,i) = f_{\mathrm{tri}}(&u_1(k,i), u_2(k,i), b, c_1, c_2, \varphi_1, \varphi_2, \\ &h_1, h_2, A_{1_1}, A'_1, A'_2, A''_1, A''_2) \end{aligned} \tag{9}$$

working on 2d photogrammetry between the corresponding pixels calculated separately for $j = 1, 2$ with

$$u_j(i,k) = f_{\mathrm{cor}}(G^{(j)}_k, i, a_x, a_y, a_0) \tag{10}$$

where $G^{(j)}_k$ is the original image information on target view k. In function f_{cor} we apply at first a 1d image operation detecting the pixel coordinates of the appropriate line i. Then we consider the co-planarity deviation given in equation 1 whereby we calculate the effects on the pixel coordinate and correct the original pixel coordinates accordingly.

Of course, the distance between the reconstructed points from the equations 8 and 9 should be minimal. Let $L_k \subset \{1, 2, \ldots, n_\mathrm{l}\}$ be the set of lines captured on target position k. Then we yield our objective function by summing all squared residues with

$$\sum_{k=1}^{n_\mathrm{v}} \sum_{i \in L_k} \|s(k,i) - r(k,i)\|^2 \to \min\,. \tag{11}$$

3.3 Linearizing and Least-Squares Adjustment

The optimization problem of equation 11 can be solved by linearizing and least-squares adjustment. But the system of equations that we get is very large and we need feasible start values as well as a workable optimization strategy. The only known values at the beginning are approximations of the calibration target. From this we reconstruct as described in [6] for every target view a set of intersection points and obtain by direct linear transformation (DLT) the inner and the outer parameter approximately. Then, the inner parameter are averaged over all target views because the sensor construction is fix.

In principle, we now have all initial values to run a linearizing and least-squares adjustment. But there are so many approximations, that a simultaneous optimization of all parameters would fail. However, we have to proceed successively. In general we start with linearizing only the outer parameter $\mathbf{p}_1$. If the optimization is converged then we proceed by subjoining the next parameter classes $\mathbf{p}_2$, $\mathbf{p}_3$ and $\mathbf{p}_4$.

Additionally, we have to exclude permanently some parameters because the outer orientation $\mathbf{p}_1$ depends directly on the calibration target parameter $\mathbf{p}_3$, $\mathbf{p}_4$, and vice versa. We yield reasonable results if we exclude the orientation of one target plane as well as the orientation of two non-parallel lines on this plane. And finally, we can only optimize the base line parameter $b \in \mathbf{p}_2$ together with the parameter set $\mathbf{p}_4$ if we note the scale of the system. We keep the scale of the system fixed by adding an extra equation considering the averaged width of all markers on the calibration target. The aimed width is given by an extra scale parameter.

4 Implementation and Results

The presentation of the following results is based on the realization of two developments: the calibration target and the stereo line scan camera system. It provides no general studies but a functional demonstration of the main approach which is implemented on a standard PC using the Eigen-library under c++.

The calibration target described in section 2.1 is made of precisely machined aluminum blocks laminated with standard prints of line pattern. It has a width of 400 mm and a height of 40 mm. The approximate values for the plane parameters $\mathbf{p}_3$ are taken from the machine presetting and the approximate values for the line parameters $\mathbf{p}_4$ we take from PDF-drawing.

The stereo line scan camera system consists of two 4k line scan cameras with a lens shift focusing the same line in a distance of about 1000 mm. With special equipment we aligned the cameras with a co-planarity error less than 0.1 pixel. The image capture is triggered and works normally for the purpose of 3d surface reconstruction with a linear unit. But due to our claim to abandon a known movement we used for the calibration process only single captured lines.

The first investigation of our method concerns the convergence behaviour. If the initial approximations of the plane and line parameter sets $\mathbf{p}_3$ and $\mathbf{p}_4$, respectively, are sufficiently precise, then our algorithm converges in less then 20 iterations. Concerning the objective function in equation 11 we get mean residues lower than 15 µm.

Of course, if there is no convergence we cannot get any results. But the question is now: Are the calculated parameter correct? To answer this question completely we need an independent measurement system which provides for all parameters at least the same accuracy as our proposed method. However, such measuring facilities we had only for the plane parameters $\mathbf{p}_3$, which we measured by a tactile system with an accuracy of 5 µm on nine different points per plane. Table 1 shows a comparison by min, max and mean deviations on each plane.

plane	1	2	3	4	Δh
mean	41.7	8.7	−27.8	64.7	21.0
min	−14.1	−23.2	−47.1	29.8	
max	82.1	11.1	−15.6	109.8	
mean	52.3	−19.2	−17.3	54.2	0.0
min	−3.5	−33.7	−36.6	19.4	
max	92.7	0.5	−5.1	99.4	

Table 1. Deviations [µm] to tactile measurement

Additionally, we compared our calibration results with the tactile measurement concerning the height of the target defined by the difference between the means of upper and lower planes. But actually a deviation in height implies a scale error which we can easily compensate by changing the scale parameter. The first part of table 1 shows results with a fixed scale parameter derived from the approximations of line parameters. The second part shows results on the same input data, but with a calculated scale parameter compensating the height error Δh. We want to note, the mean residues for the underlying investigation are 14.52 µm, which is not a bad value for systems of this size.

An indirect, but more comparable investigation of our calibration method is based on 3d surface measurements. Of course, such investigation depends on the 3d measuring method itself, but normally the absolute measurement errors are caused by an inaccurate calibration. However, as shown in table 2 we investigate our method according to the technical guideline "VDI/VDE 2634-2" where we operate on three proposals: the maximum deviation from the exactly known surface of a sphere and a plane, respectively, as well as the length deviation between the centers of two spheres connected by a bar of exactly known length.

investigation	our method	Sun et.al. [9]
sphere deviation [µm]	249.4	454.0
plane deviation [µm]	179.2	229.0
lenght deviation [µm]	62.2	–

Table 2. Investigation by 3d measurement

Furthermore, we contrast the maximal deviation results with the quite typical results given in [9]. Of course, both results are based on different setups and different reference surfaces, but we can reason that our proposed calibration is at least as good, especially because our measuring field is with 400 mm almost twice as large.

5 Conclusion

We proposed a new calibration method for stereo line scan camera systems whereby we defined different parameter classes and an approach working on it. By using point residues within the viewing plane we defined an objective function which is applicable by linearizing and least-squares adjustment. Our optimization approach converges well and although we abandon on linear movement and a precisely known calibration target our results are comparable or better then other line scan calibration methods using such technical requirements. However, this leads to a more general applicability of line scan cameras in the field of computer vision.

Acknowledgements: This work was supported by the German Federal Ministry of Education and Research under Grant 03ZZ0425G.

References

[1] Azra Fetic, Davor Juric, and Dinko Osmankovic. The procedure of a camera calibration using camera calibration toolbox for matlab. In *MIPRO, Proc. of the 35th International Convention*, pages 1752–1757, 2012.

[2] Simon X Godber and Max Robinson. Machine vision using line-scan sensors. In *Proc. SPIE 1823, Machine Vision Applications, Architectures, and Systems Integration*, pages 114–130, 1992.

[3] Bingwei Hui, Gongjian Wen, Zhuxin Zhao, and Deren Li. Line-scan camera calibration in close-range photogrammetry. *Opt. Engineers*, 51(5):053602–1, 2012.

[4] Trendafil Ilchev, Erik Lilienblum, Bernd Joedicke, Bernd Michaelis, and Markus Schnitzlein. A stereo line sensor system to high speed capturing of surfaces in color and 3d shape. In *Proc. of IVAPP*, pages 809–812, 2012.

[5] Erik Lilienblum and Ayoub Al-Hamadi. A structured light approach for 3-d surface reconstruction with a stereo line-scan system. *IEEE Trans. on Instrumentation and Measurement*, 64(5):1266–1274, 2015.

[6] Erik Lilienblum, Ayoub Al-Hamadi, and Bernd Michaelis. A coded 3d calibration method for line-scan cameras. In *Pattern Recognition, Proc. of GCPR 2013*, LNCS 4263, pages 81–90. Springer, 2013.

[7] Carlos A Luna, Manuel Mazo, Jose Luis Lazaro, and Juan F Vazquez. Calibration of line-scan cameras. *IEEE Trans. on Instrumentation and Measurement*, 59(8):2185–2190, 2010.

[8] Fabio Remondino and Clive Fraser. Digital camera calibration methods: considerations and comparisons. *Int. Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(5):266–272, 2006.

[9] Bo Sun, Jigui Zhu, Linghui Yang, Shourui Yang, and Yin Guo. Sensor for in-motion continuous 3d shape measurement based on dual line-scan cameras. *Sensors*, 16(11):1949, 2016.