

SOFTWARE APPLICATION FOR CALIBRATION OF STEREOSCOPIC CAMERA SETUPS

Dariusz Rzeszotarski, Paweł Pelczyński

*Technical University of Łódź, Institute of Electronics, Wólczajska 211/215, 90-924 Łódź, Poland
(dariusz.rzeszotarski@p.lodz.pl, pawel.pelczynski@p.lodz.pl, +48 42 631 2617)*

Abstract

The article describes an application for calibration of a stereovision camera setup constructed for the needs of an electronic travel aid for the blind. The application can be used to calibrate any stereovision system consisting of two DirectShow compatible cameras using a reference checkerboard of known dimensions. A method for experimental verification of the correctness of the calibration is also presented. The developed software is intended for calibration of mobile stereovision systems that focus mainly on obstacle detection.

Keywords: camera calibration, canonical stereovision system, intrinsic and extrinsic parameters.

© 2012 Polish Academy of Sciences. All rights reserved

1. Introduction

Proper calibration of the camera setup is an important step during construction of any stereovision system. A typical stereovision setup consists of two cameras with parallel optical axes, separated by a narrow base B [1]. Its main use is the determination of depth in a registered 3D scenes in the form of disparity [1] – the shift between observed details in the left and right images. To be able to utilize fast block matching methods for disparity estimation [3] it is necessary to satisfy a number of conditions: both cameras should have identical pin-hole model parameters, their optical axes should be perfectly parallel and the image matrices should lie in the same plane. When all three conditions are met, corresponding rows of pixels in both images should be collinear [1]; however, satisfying these conditions is nearly impossible in real cameras, due to distortion introduced by the camera optics and imperfections in the camera positioning. Instead of expensive corrections of the optical and mechanical properties of the stereovision system, digital image adjustment is commonly used to satisfy the aforementioned model requirements [4].

At the Institute of Electronics, Lodz University of Technology, an application was developed to quickly estimate the distortion parameters introduced by the optics and the stereovision system's parameters, in order to correct the registered images digitally. The software utilizes known techniques for stereovision system identification [5]. The original achievement of the authors is the application's universality, allowing to use the software with any pair of cameras that use DirectShow drivers, and a user-friendly interface that allows for fast stereo system calibration and rectification.

The developed application for camera calibration has three main capabilities: calibration of intrinsic parameters of each camera, calibration of a two-camera system for axis alignment, and rectification of the stereo images. The application can connect to PointGrey Flea [1] cameras through their specific API or with any cameras detected in the system as DirectShow

[7] visual inputs. The calibration utilizes algorithms provided in the OpenCV [5] library. Below is the description of the utilized camera model and calibration algorithms.

2. Camera Model

The task of the calibration procedure is to determine the intrinsic and extrinsic camera model parameters. The intrinsic parameters describe the relation between image pixel coordinates x_p, y_p and the spatial coordinates X, Y, Z :

$$Z \begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = K \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}, K = \begin{bmatrix} f/h_x & 0 & c_x \\ 0 & f/h_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, \quad (1)$$

where:

- K – the camera projection matrix;
- f – focal length of the camera's optics;
- h_x, h_y – size of a single pixel along the x and y axis, respectively, of a two-dimensional frame of reference tied to the image detector plane;
- c_x, c_y – coordinates of the Z axis intersection with the detection plane.

The camera's optical system may introduce geometric distortions. Their presence can be modeled by the following equations:

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = \left(1 + k_1 r^2 + k_2 r^4\right) \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \Delta, \quad r^2 = x_n^2 + y_n^2, \\ \Delta = \begin{bmatrix} 2p_1 x_n y_n + p_2 (r^2 + 2x_n^2) \\ 2p_2 x_n y_n + p_1 (r^2 + 2y_n^2) \end{bmatrix}, \quad x_n = \frac{X}{Z}, y_n = \frac{Y}{Z}, \quad (2)$$

where:

- k_1, k_2 – coefficients responsible for radial distortions;
- p_1, p_2 – tangential distortions;
- r – distance of an undistorted point from the focal center;
- x_n, y_n – normalized undistorted coordinates;
- x_d, y_d – distorted coordinates.

In the case when point P' in space is in a different coordinate system than the camera, the relation between the coordinates $P = [X, Y, Z]^T$ in the camera's system and the coordinates $P' = [X', Y', Z']^T$ can be written as:

$$P = RP' + T, \quad (3)$$

where R is the rotation matrix and T is the translation vector. Parameters: $f, c_x, c_y, k_1, k_2, p_1, p_2$ are intrinsic camera parameters, while R and T are the extrinsic parameters.

3. Stereovision camera system

In the stereovision camera system (Fig. 1) one of the cameras (the left one by default) is treated as the reference camera. In practice, this means that the reconstructed spatial points are expressed using coordinates connected with the reference camera.

The coordinates of a point $P = [X, Y, Z]^T$ in the reference camera's coordinate system, and the coordinates of that point in the second camera's coordinate system $P' = [X', Y', Z']^T$ are connected using the formula:

$$P = RP' + T. \quad (4)$$

A point $P_w=[X_w, Y_w, Z_w]^T$ in an external coordinate system is related to the two camera coordinate systems using the following formulas:

$$P = R_w P_w + T_w, \quad P' = R'_w P_w + T'_w. \quad (5)$$

Knowing the relative position of the external coordinate system and the camera coordinate systems, the matrices of rotation R and translation T can be calculated:

$$R = R_w R'_w{}^T, \quad T = T_w - R_w R'_w{}^T T'_w. \quad (6)$$

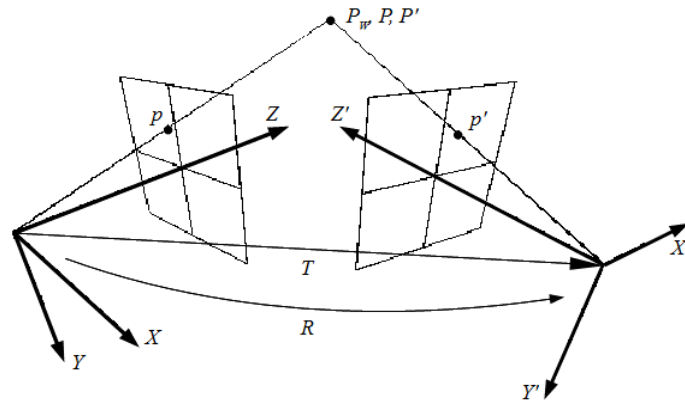


Fig. 1. A stereovision camera system.

4. Calibration of a single camera

An important element of the calibration procedure is a reference object of known shape and dimensions, containing characteristic points in known orientation. For the calibration of the cameras in the described application a flat 7x7 checkerboard is used, with 60mm square fields, the corners of which are the characteristic points (Fig. 2). The calibration object is shown in its entirety to the calibrated camera(s) in a number of positions. Knowing the dimensions of the reference object, the coordinates of the characteristic points can be determined in the coordinate system tied to the object. In the case of the checkerboard, the X and Y axes run along its edges, while the Z axis is perpendicular to its surface (Fig. 2).

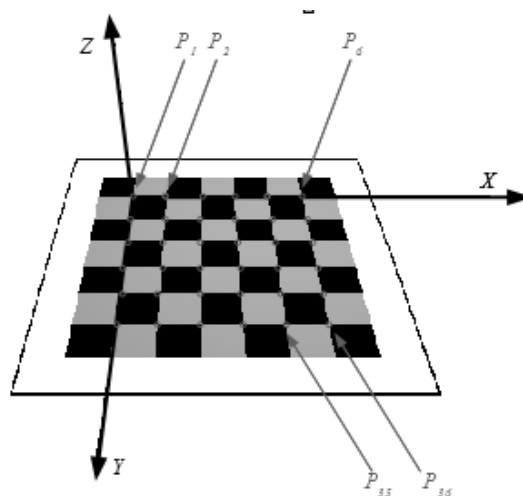


Fig. 2. The calibration checkerboard with its coordinate system and characteristic points $P_1, P_2, \dots, P_6, \dots, P_{35}, P_{36}$.

The calibration procedure consists of the following steps:

1. Find the image coordinates of the corners found on the checkerboard for a number of positions of the board relative to the camera system [8].
2. For each of the positions determine the homography matrix for the transposition of the coordinates of points on the surface of the board and the coordinates in the image plane.
3. Basing on the found matrices estimate the intrinsic and extrinsic camera parameters.
4. Optimize the parameters through minimization of the difference between the known and calculated coordinates of the current model [5].

5. Calibration of the stereovision camera system

When the intrinsic parameters of both cameras are known, the calibration of the stereovision system consists of the computation of the rotation and translation matrices between them according to formula (4). The calibration of the extrinsic parameters R and T consists of determining the two homography matrices: first homography H between the calibration board surface and reference image surface in the first camera and the second homography H' between calibration board surface and corresponding surface in the second camera.

$$\lambda_i \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = H \begin{bmatrix} X_i \\ Y_i \\ 1 \end{bmatrix}, \quad \lambda'_i \begin{bmatrix} x'_i \\ y'_i \\ 1 \end{bmatrix} = H' \begin{bmatrix} X_i \\ Y_i \\ 1 \end{bmatrix}, \quad (7)$$

where: λ and λ'_i □ scaling factors.

Basing on the homography matrices H, H' , rotation matrices (R, R') and translation vectors (T, T') between the coordinate systems of the cameras and the coordinate system tied to the reference checkerboard (5) are calculated. Using formulas (6) the extrinsic parameters, *i.e.* the R and T matrices, between the camera pair can be calculated.

6. Canonical stereovision system

A stereovision system is considered to be canonical if the epipolar lines in the two cameras are parallel and the equation for an epipolar line for a given point $p' = [x_p, y_p]^T$ in one of the cameras is $y_p = y'_p$. In other words, in a canonical system pairs of corresponding image points representing the same point in space will lie on parallel lines. A camera system will be canonical if the axes of the coordinate systems of the two cameras are parallel and translated only along one axis *e.g.* X and the intrinsic camera parameters differ only in the coordinates of the central point $c_x \neq c'_x$ (assuming separation along the X axis).

6.1. 3D reconstruction basing on a pair of images from a canonical camera system

The reconstruction matrix is a 4x4 array that describes the relation between a spatial point $P = [X, Y, Z]^T$, the image point $p = [x, y]^T$ in the reference camera of a canonical system, and the disparity d , in the following manner:

$$W \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -c_x \\ 0 & 1 & 0 & -c_y \\ 0 & 0 & 0 & f \\ 0 & 0 & \frac{1}{fT_x} & \frac{c'_x - c_x}{fT_x} \end{bmatrix} \begin{bmatrix} x \\ y \\ d \\ 1 \end{bmatrix}. \quad (8)$$

6.2. Rectification of the stereovision system

In practice, real stereovision systems are not canonical. In order to enable an easy and fast reconstruction process, the images from the cameras must undergo mapping to a canonical form. The process of mapping the stereovision images is called rectification. The rectified images are used in the implemented reconstruction procedure in the following manner: for each pixel from the reference image a corresponding pixel is searched for in the second image, i.e. the disparity of the horizontal coordinates of all the pixels is calculated; the end result of this process is called a dense disparity map, which can be translated into a cloud of spatial points using (8).

The stereovision system before rectification has a rotation matrix R , a translation vector T , projection matrices K and K' , as well as parameters describing the geometrical distortions of the reference camera k_1, k_2, p_1, p_2 and the second camera k'_1, k'_2, p'_1, p'_2 .

The task of the rectification algorithm is to find such rotation matrices $R_{||0}, R_{||1}$ for each camera that using them the system's rotation matrix $R_{||}$ leads to a unity matrix and the translation vector $T_{||}$ has only an X component. The rectification procedure was implemented using a matrix of the target image size (after rectification), the elements of which (with coordinates $x_{p||}, y_{p||}$) store image pixel coordinates x_p, y_p from the image before rectification. The rectification procedure uses the said matrix to assign pixel $p_{||}(x_{p||}, y_{p||})$ values of pixels pointed by the matrix $p(x_p, y_p)$. Because the coordinates x_p, y_p stored in the matrix are not integer numbers, the rectification procedure uses bilinear interpolation to more precisely determine the value of the pixels pointed at by x_p, y_p in the source image (before rectification).

7. Software description

The application for calibration of two camera stereovision systems was developed in the Borland Builder C++ environment with the use of the OpenCV library [5]. The software consists of the following modules:

- Graphical interface:
 - Main application window; class TFormMain (file: MainForm.cpp/ MainForm.h);
 - Additional settings, calibration parameters for input images; class TFormControl (file: ControlForm.cpp / ControlForm.h).
- Thread responsible for reading and processing the images from the cameras – class TGrabThread (file: GrabThread.cpp / GrabThread.h)
- Camera calibration, saving/loading calibration parameters – class CalibFilter (file: FlyCalibFilter.cpp/ FlyCalibFilter.h)

The class TFormMain contains functions responsible for: detecting cameras connected to the computer, allocating memory buffers for the input images, saving/loading settings connected with the input images, the calibration procedure, starting/stopping image acquisition, processing and displaying the images, starting/stopping the calibration procedure.

The TGrabThread class contains the main function of the thread, which loops one of the following methods:

- FreeRunning() – during acquisition, processing and displaying of the input images. The image acquisition is handled by the function GrabImages(). Initial processing of the images depending on the *Preprocessing* options chosen in the main window: correction of geometric distortions introduced by the camera optics and rectification of a pair of images. The images are binarized using a standard or an adaptive threshold filter according to parameters *Search Corners Parameters* in the main application window. Image pairs can be matched according to an algorithm turned on in the *Find Correspondence* option.

- Calibrate() – during camera calibration. The images for calibration are also acquired using the GrabImages() function. They are analyzed using the FindEtalon() function that localizes the corners of the squares on the reference checkerboard. Once all $(M-1) \times (N-1)$ vertices are found on an $M \times N$ checkerboard, the function FindEtalon() returns the true value and the coordinates of found corners for the left and right camera for a given j -th position of the checkerboard are stored using a Push() function. In the case when all $(M-1) \times (N-1)$ vertices are found for K (preferably different) positions of the calibration board, further acquisition of images by the FindEtalon function is stopped and the calculated image coordinates are used for the previously described calibration function.

Upon startup, the application searches for any PointGrey Flea cameras connected to the computer, as well as for any cameras seen as DirectShow® filters by the system. If both types of cameras are detected, the PointGrey Flea cameras are used by default. A screenshot of the main application is shown on Fig. 3.

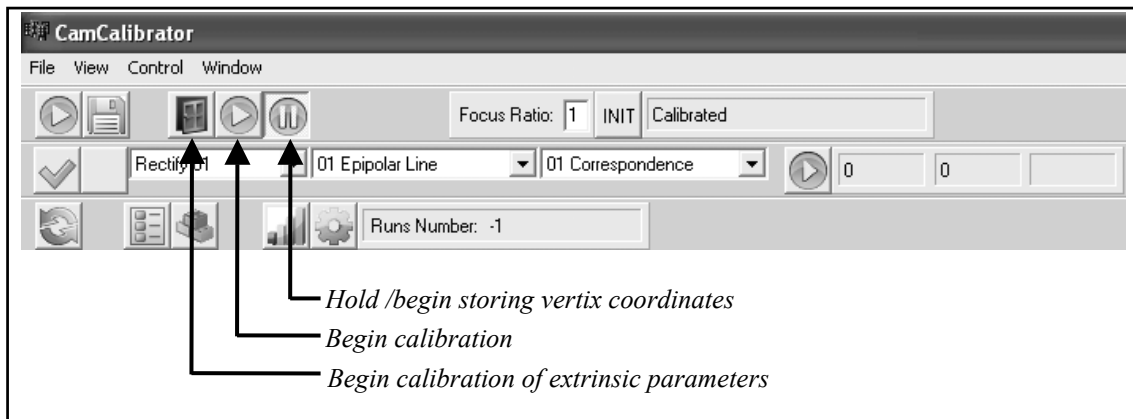


Fig. 3. Main toolbar of the application.

Parameters connected with the algorithm that searches for image coordinates of the reference checkerboard corners are collected in the main window's panel *Search Corners Parameters* (Fig. 4). Parameters of the thresholding filter are listed as *Par. 1*, ... *Par. 3* and their meaning is dependent on the chosen filter type (*Adaptive/Nonadaptive Threshold*). Coordinates of the vertices are located in a binary image, which is the result of the thresholding filter of the chosen type and with chosen parameters. The binarization filter can be inverted by choosing the *Inv/Noninv Threshold* option. After binarization, the object undergoes morphological operations if the *Dilation/Erosion* is checked. The order of locating the checkerboard vertices can be changed using the: *Change Order 0*, *Change Order 1*, *Change Order 2* options for each camera. All of the previously mentioned parameters and options can be changed for each of the camera separately; in order to do so, a camera must be selected using the index of camera tab.

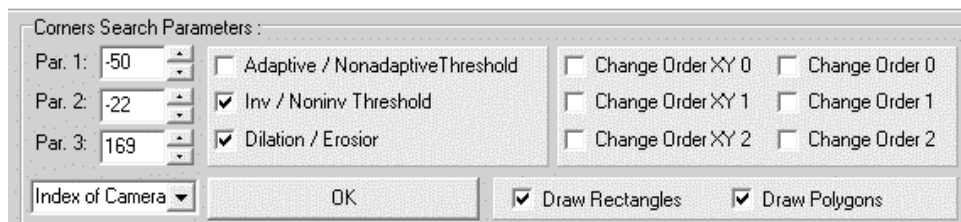


Fig. 4. Parameters connected with the algorithm that searches for image coordinates of the reference checkerboard corners.

8. Calibration of intrinsic and extrinsic parameters

8.1. Calibration of intrinsic parameters

In order to begin the calibration of the camera system with regard to its intrinsic and extrinsic parameters, the button “Begin Calibration” must be pressed. Two windows are displayed for every camera: one containing the original image and a second containing a thresholded binary image (Fig. 7). The binary image allows to better select the parameters of the algorithm that searches for the reference checkerboard’s vertices.

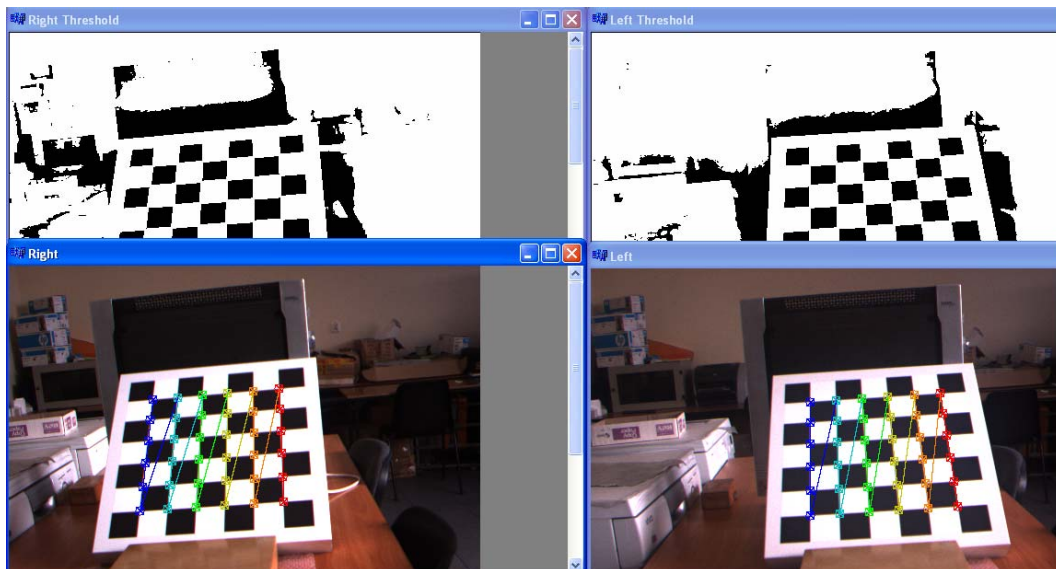


Fig. 5. The images from a stereovision camera in the calibration procedure.

The algorithm of the calibration procedure does not store the coordinates of the detected vertices until both of two conditions are met: all vertices have been detected in both cameras, and the button “begin/hold storing vertex coordinates” has been pressed. After the coordinates of the vertices have been stored for all cameras, the procedure is repeated and the reference board should be placed in a different position that is still visible to both cameras. While the position is being changed, the button “hold storing vertex coordinates” can be pressed so the same set of coordinates is not accidentally acquired for a second time. The calibration procedure ends when the coordinates were stored for a number of positions specified by the *Total Numbers of Frames to Calibrate* parameter in the *Calibration Grid* panel in the *Calibration Parameters* tab.

8.2. Calibration of extrinsic parameters

After full calibration, if the respective positions of the cameras were changed, it is sufficient to perform only the calibration of the extrinsic parameters. To begin it, the button “Begin calibration of extrinsic parameters” must be pressed. The vertices of the reference checkerboard are then found as many times as it is specified in the *Calibration Grid* tab by the parameter *Number of Frames to Calibrate Extrinsic*.

9. Experimental verification

The camera system was calibrated using the developed application. In the first stage, each camera was calibrated separately for its intrinsic parameters. The reference object was

observed in 15 different positions for each calibrated camera. In the second stage, the stereovision system was calibrated using 4 different frames containing the checkerboard.

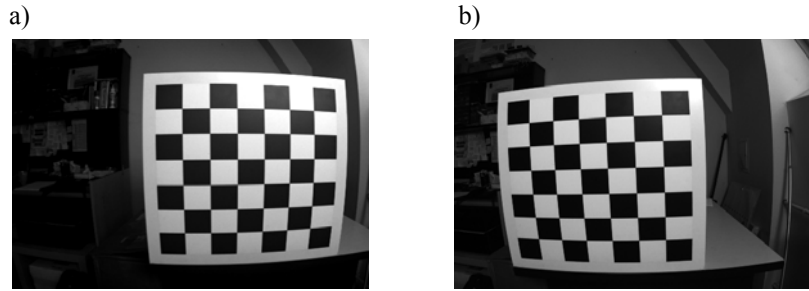


Fig. 6. A pair of images obtained without distortion correction and rectification: a) left camera; b) right camera.

Quantitative measurements of the correctness of the calibration and rectification procedures were made. In order to perform them a measurement setup was prepared which consisted of a pair of cameras in fixed positions and a reference checkerboard, that was moved on a flat surface with 1mm precision. The board underwent only translations and no rotations, while the cameras remained fixed throughout the procedure. Two color PointGrey Flea cameras with 1/3” matrices were fixed on a rigid frame on a steel tripod. The cameras had 3.5mm fixed focal length optics and a FireWire interface to the PC. The short focal length of the lenses caused heavy radial distortions, as seen in Fig. 6. The parameters obtained in the calibration procedure are presented below in Table 1.

Table 1. Camera intrinsic parameters.

Parameter	Left camera	Right camera
Image Size	512x384	512x384
x Focal Point	394.9708895887	396.0624713925
y Focal Point	395.2669252887	396.6943544664
x Principal Point	288.4963122808	266.3764681239
y Principal Point	186.3495404384	188.6713642335

The table shows that the image matrices of both cameras have nearly square pixels and mean focal lengths of 395.1189 pix and 396.3784 pix. The small difference is the result of manual focusing of the cameras. The parameters of the camera distortions are presented in Table 2.

Table 2. Camera distortion parameters.

Distortion Coefficients	Left camera	Right camera
k1	-0.3677287909	-0.3767188981
k2	0.2060184194	0.2414276010
p1	-0.0004856769	-0.0003803136
p2	0.0014054880	0.0011410230

The parameters of the relative rotation and translation between the cameras are presented in the form of a rotation matrix R and a translation vector T :

$$R = \begin{bmatrix} 0.9984626089 & -0.0278902866 & -0.0479014675 \\ 0.0267947641 & 0.9993679224 & -0.0233622846 \\ 0.0485227709 & 0.0220428591 & 0.9985788167 \end{bmatrix}, T = \begin{bmatrix} 221.0625940786 \\ -2.2512929191 \\ 4.3350166837 \end{bmatrix}. \quad (9)$$

The base B calculated from the above parameters, *i.e.* the distance between the optical axes after rectification, is equal to 221.1166 mm.

Verification consisted of measuring:

- the inaccuracy of estimation of the position of the reference checkerboard for several distances from the stereovision setup;
- the distortions of the square fields of the reference board in the left and right image;
- the differences of the y coordinate in the images of corresponding points in the left and right images.

The measurement was made for three positions of the reference board, translated away from the cameras by 500 mm (Fig. 7). The tested setup had no precise tools for measuring the position and orientation of the cameras, so only relative errors of the position of characteristic points (vertices) of the reference checkerboard were measured. The reference values consisted of the known dimensions of the reference object and its translational movements.

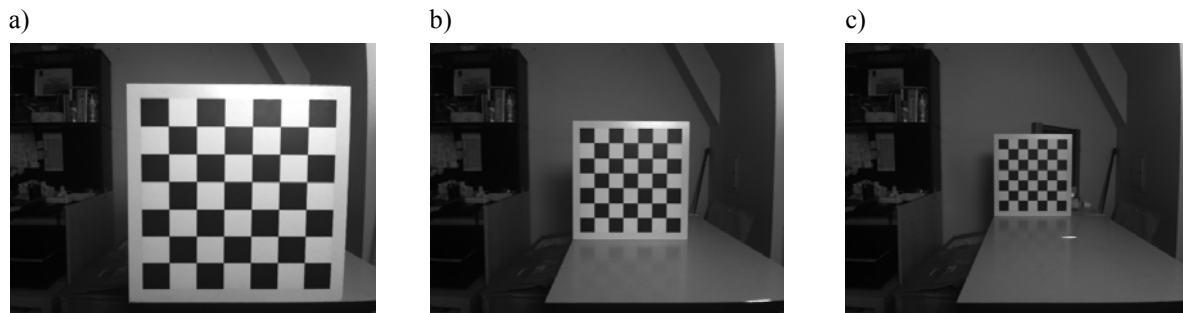


Fig. 7. Right image of calibration board after distortion correction and rectification: a) the closest checkerboard position; b) 0.5m shift of the checkerboard; c) 1.0 m shift of the checkerboard.

The procedure determining the vertices was included in the OpenCV library [5] and consisted of the following stages: binarization of the image, detection of all quadrangles using closed contour methods, and localization of all vertices of the checkerboard. Reconstruction of the spatial positions of the vertices was performed through standard stereoscopic triangulation described by formula (8).

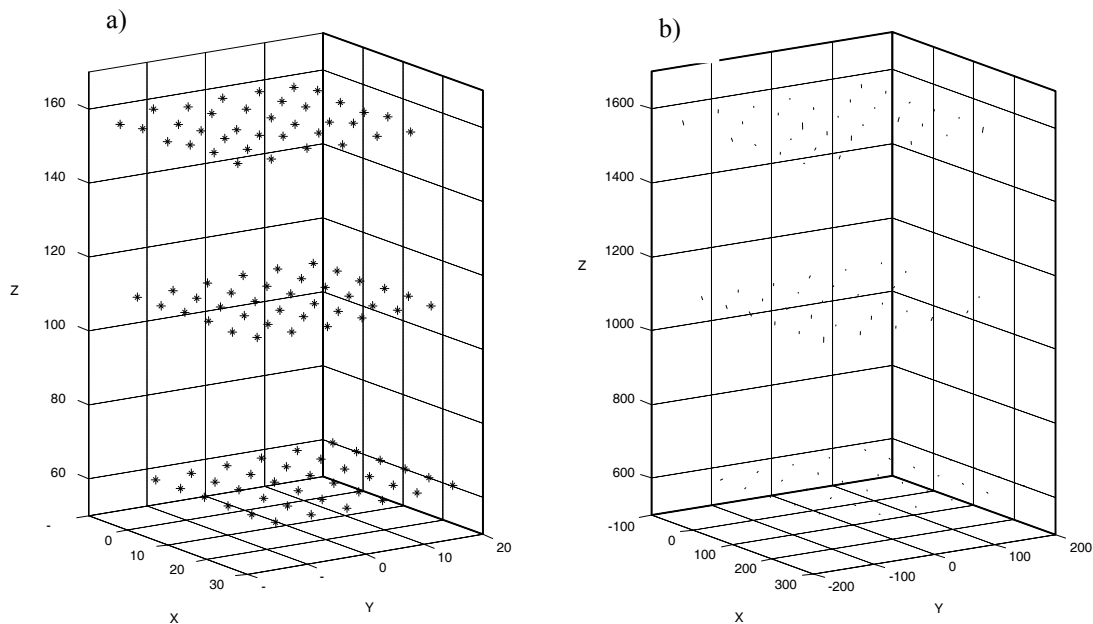


Fig. 8. Corner locations and their position estimation errors in 3D: a) 3D locations of checkerboard corners for the three board positions; b) errors of 3D location estimation of checkerboard corners.

A spatial model of the vertex mesh was prepared for three positions of the reference board, which was then fit to the measurement data in order to calculate the accuracy of

reconstruction. The fitting consisted of computing such a translation vector and a rotation matrix that minimized the mean square error of the reconstruction of the vertices as shown in Fig. 8a. The determined inaccuracies are presented in Fig. 8b as lines connecting the fitted and measured spatial coordinates of the vertices.

The largest differences were found for the furthest position of the reference board ($Z=1550$ mm), which is the result of the limitations of narrow base stereoscopy.

The measured average and maximum errors for each of the three positions of the reference board are given in Table 3. The 3D reconstruction errors are influenced by the accuracy of calibration, the detection of the vertices and the stereoscopic triangulation.

Table 3. Average and maximum errors of corner position estimation in 3D.

	1. position	2. position	3. position	mean
Average error [mm]	2.8670	4.7713	5.7604	4.4663
Maximum error [mm]	4.4662	14.3150	16.6295	16.6295

As expected, the errors grow as the distance from the cameras increases. This is caused by the limited resolution of the acquired images, which in turn limits the accuracy of localization of the vertices' positions in the images [8]. Small errors for the first position of the checkerboard allow the calibration and rectification methods utilized to be deemed effective and sufficient for application in computer vision systems [10].

The average deviation (measured in pixels) between the detected vertices and a projection of their model on the virtual images was used as the measure of the distortion of the square mesh in the rectified images. Fig. 9a shows the estimated locations of the vertices in all three positions of the reference board, while Fig. 9b shows the errors visualized as lines connecting the detected and projected positions. The average and maximum errors are summarized in Table 4.

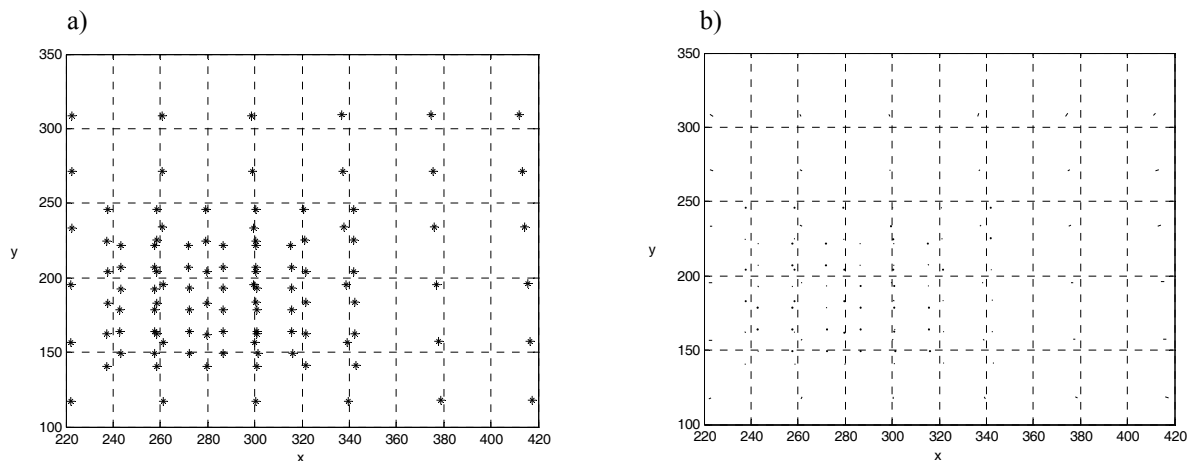


Fig. 9. Estimated locations of the vertices of the reference board and their estimation errors: a) estimated locations of the vertices in all three positions of the reference board; b) errors visualized as lines connecting the detected and projected positions (they are almost invisible).

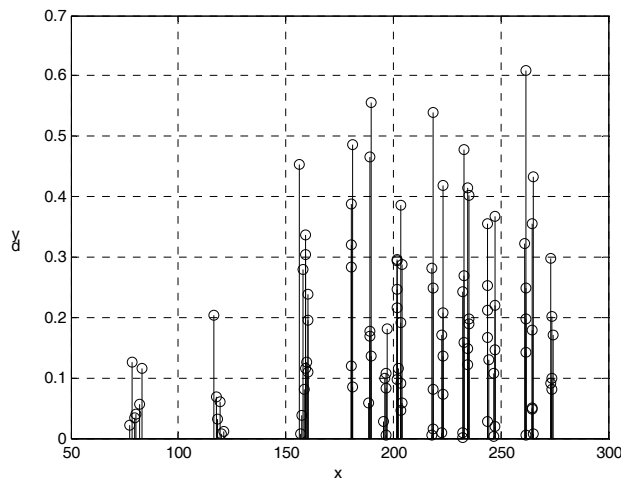
The low values of the errors allow to draw the conclusion that the output images have low geometric distortion allowing for correct stereovision reconstruction [8].

The correctness of the rectification of the stereoscopic system is measured by the accuracy of the vertical localization of points in the left and right image corresponding to the same spatial point. The y coordinate differences were calculated for all vertices in the three reference board positions. The distribution of the absolute error values is visualized in Fig. 10.

Table 4. Average and maximum errors in vertex image locations.

	Left camera	Right camera
Average error [pix]	0.5351	0.6683
Maximum error [pix]	1.8539	2.3564

The average error Δy for all vertices was equal to 0.1804[pix], and the maximum error was 0.6096 [pix]. Since the average error between the y coordinates is less than 0.2 of a pixel, fast algorithms for block matching can be effectively implemented [8].

Fig. 10. Differences of the y coordinates of corners in the left and right image for varying x .

10. Conclusions

The paper presents an application for calibration of stereovision camera systems. The theoretical basics of the calibration process were shortly described as well as the techniques used in the implementation. Results of the verification of the calibration using a sample stereovision system were also presented. The achieved results allow to use the developed application to calibrate stereovision two-camera systems used for computer and robot vision [11], especially for mobile systems with a narrow base [12]. The application is currently utilized for calibration of mobile stereovision systems in electronic travel aids for the blind [13].

Acknowledgements

This work has been supported by the Ministry of Science and Higher Education of Poland research grant no. NR02-0083-10 in years 2010-2013.

References

- [1] Ma, Y., Soatto, S., Kosecka, J., Sastry, S. (2003). *An invitation to 3D vision, from images to models*. Springer Verlag.
- [2] Cyganek, B. (2002). *Computer processing of 3D images*. Akademicka Oficyna Wydawnicza EXIT, Warsaw. (in Polish)
- [3] Brown, M.Z., Burschka, D., Hager, G.D. (2003). Advances in computational stereo. *IEEE Trans. pattern analysis and machine intelligence*, 25(8), 993-1008.

- [4] Fusiello, A., Trucco, E., Verri, A. (2002). A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications* 2002, 12, 16-22.
- [5] Gary Bradski. G., Kaehler, A. (2008). *Learning OpenCV*. O'Reilly Media.
- [6] <http://www.ptgrey.com/>
- [7] Pesce, M.D. (2003). *Programming Microsoft® DirectShow® for Digital Video and Television*. Microsoft Press.
- [8] Jendernalik, W., Jakusz, J., Blakiewicz, G., Piotrowski, R., Szczepański, S. (2011). CMOS realisation of analogue processor for early vision processing. *Bull. Pol. Acad. Sci-Tech., Sci.*, 59, (2), 141-147.
- [9] Cyganek, B., Siebert, J.P. (2009). *An introduction to 3D computer vision techniques and algorithms*. John Wiley & Sons, Ltd.
- [10] Skulimowski, P., Bujacz, M., Strumiłło, P. (2009). Detection and Parameter Estimation of Objects in a 3D Scene. *Image Processing & Communications Challenges, Academy Publishing House EXIT*, Warsaw.
- [11] Dutkiewicz, P., Kielczewski, M., Kozłowski, K., Pazderski, D. (2010). Vision localization system for mobile robot with velocities and acceleration estimator. *Bull. Pol. Acad. Sci-Tech., Sci.*, 58(1), 29-42.
- [12] Skulimowski P., Strumiłło P. (Sept. 3-7, 2007). Obstacle Localization in 3D Scenes from Stereoscopic Sequences. *15th European Signal Processing Conference (EUSIPCO 2007)*, Poznan, 2095-2099.
- [13] Bujacz M., Skulimowski P., Strumiłło P. (Sept. 2012). Naviton – a prototype mobility aid for auditory presentation of 3D scenes. *Journal of Audio Engineering Society*, 60(9), 696-708.