Color based Passpoint Identification for Calibration of Stereo-Cameras

Udo Ahlvers, Udo Zölzer, Björn Keck, René Opitz
Department for Signal Processing and Communications
Helmut-Schmidt-University / University of the Federal Armed Forces
Hamburg, Germany
`http://ant.hsu-hh.de/ahlvers`
`udo.ahlvers@hsu-hh.de.de`

Abstract. The calibration of cameras is an important part in every computer vision system. With the use of passpoints, their coordinates both in the 3D scene as well as in the 2D image plane have to be determined. In order to automate their localisation and identification in the image, a color based approach is presented in this paper. First, the localisation is performed using the intensity component. After grouping the points in each plane, the identification is then based on a separation in the Lab color space. A suitable 3D passpoint frame has been constructed for this task. In opposite to prior solutions, our approach needs neither user interaction nor complicated passpoint textures. The algorithm has been integrated and applied successfully in a graphical user interface for stereoscopic image processing, where it is necessary as a preprocessing step to ensure standard stereo geometry.

1. Introduction

In real vision systems, the determination of different camera parameters (inner and outer orientation) is an essential preprocessing step. For stereoscopic camera setups, this calibration procedure has to be performed independently for both cameras. The following rectification, i.e. the reprojection of both images into a common virtual plane ensures parallel optical axes [1]. This significantly simplifies all subsequent stereoscopic processing, as corresponding image poins now lie in the same row in both images (epipolar constraint).

In the past, several methods have been proposed for geometric camera calibration, e.g. [2]. Despite the existence of self-calibration techniques in photogrammetry, most setups in computer vision still make use of so-called $passpoints$, which are located on a 3D passpoint frame. The common principle is to evaluate the projection of known spatial coordinates into image coordinates during the calibration step. For grayscale images, these 2D coordinates had to be determined interactive and user-supported [3]. In this paper, we present a fully automatic approach based on color information.

In Section 2, the basics of computer vision are given. The idea of automatic color based passpoint identification is presented in Section 3. Results of our apporach are discussed in Section 4. Finally, a conclusion and an outlook for future work are given.

2. Computer Vision Fundamentals

Figure 1 shows the basic geometrical setup in a computer vision system [3].

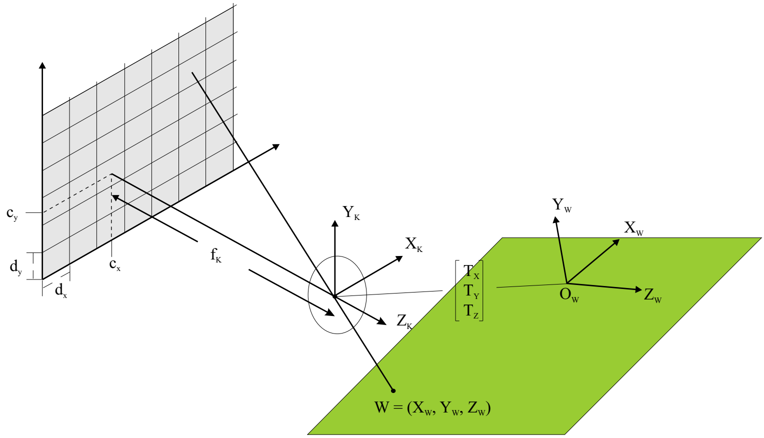


Figure 1: Interrelation between different coordinate systems (from [3]).

Typically, there are three different coordinate systems defined:

- 3D world coordinates (X_W, Y_W, Z_W)
- 3D camera coordinates (X_K, Y_K, Z_K)
- 2D image coordinates (x_b, y_b) .

To make the distinction clear, capital letters are used for the 3D coordinate systems. Further separation is performed by corresponding subscripts W, K, respectively. The 2D image coordinates are described with small letters. Generally spoken, Figure 1 describes a perjective projection from the 3D world into the 2D image plane by

$$\begin{pmatrix} x \\ y \\ h \end{pmatrix} = P \cdot \begin{pmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{pmatrix}, \quad x_b = \frac{x}{h}, \quad y_b = \frac{y}{h}, \tag{1}$$

wherein homogeneous coordinates are used usually [1]. The $3\text{x}4$ projection matrix P contains the intrinsic and extrinsic parameters for one camera, namely

- Projection main point c_x, c_y
- Camera constant f_K in $[px]$
- Translation parameters T_X, T_Y, T_Z

- Orientation parameters $r_1 \dots r_9$.

A common tool to determine these parameters is the Direct Linear Transform (DLT). It was first presented in 1971 and has since then been widely used [1, 2, 3]. Herein only linear equations have to be solved, omitting nonlinar optimization algorithms. With that, Eq. 1 can be written as

$$\begin{pmatrix} x \\ y \\ h \end{pmatrix} = \begin{pmatrix} f_x & 0 & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} r_1 & r_2 & r_3 & T_X \\ r_4 & r_5 & r_6 & T_Y \\ r_7 & r_8 & r_9 & T_Z \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{pmatrix} . \qquad (2)$$

The Matrix P contains 11 unknowns (because $P(3,4) = 1$ in homogenious representation). For each passpoint (see Section 3) we get two equations, one for each image coordinate x_b, y_b. This results in a super-conditioned linear equations systems, which can be solved easily. For more details on solving this system we refer to [4].
When dealing with a stereoscopic camera setup, the projection matrix P has now to be determined for *both cameras*. Afterwards the images have to be arranged (i.e. *rectified*) is such a way as if they had been captured by a setup adjusted to standard stereo geometry. Figure 2 shows the principle of this reprojection step.

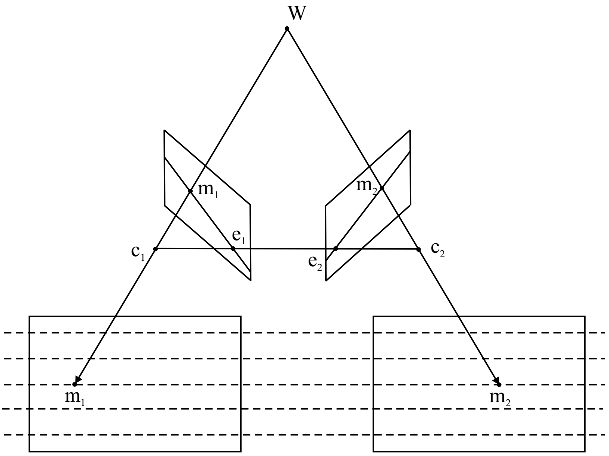
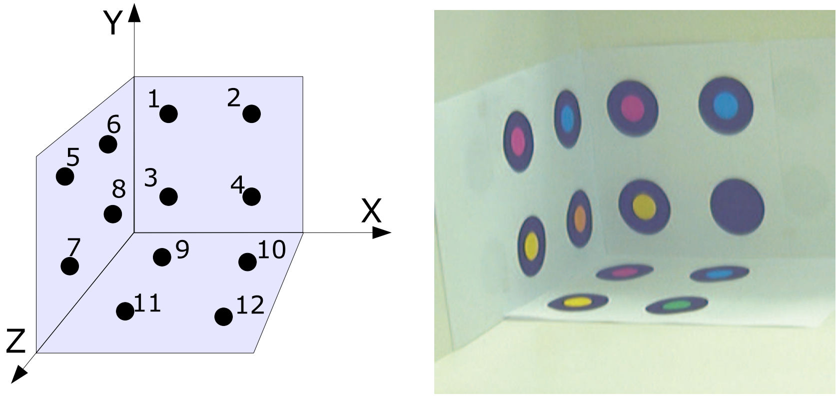


Figure 2: Rectification into parallel image planes by reprojection.

The rectification is now performed by deriving new projection matrices $P_{rect,L}, P_{rect,R}$ from the already known ones. These new matrices are defined by a rotation around the projection centres c_L and c_R to create the new virtual parallel image planes. This approach was realised by an algorithm proposed by [5].
After this final rectification step, standard stereo geometry has been ensured. This leads to a simplification for various subsequent stereo vision applications, e.g. disparity estimation [6], coding and visualisation [7].

3. Color based Passpoint Identification

One of the keypoints in the calibration procedure is the measurement of passpoint locations in image coordinates (x_b, y_b). When working with grayscale images, this had to be done either interactive and user-supported [3], or passpoints with different design and textures had to be used, forcing a complicated pattern recognition process. We now propose a fully automatic solution for passpoint location *and* identification based on color information. The corresponding hardware is shown in Figure 3.



(a) (b)

Figure 3: 3D passpoint frame. (a) Passpoint numbers, (b) associated colors.

The constructed 3D frame (a) contains 12 passpoints altogether, distributed over three planes. Each plane contains a unique color (back = black, left = orange, bottom = green) based on [8]. Furthermore, the points are surrounded by a black circle (b) to enhance the contract against the background [9].

First, the captured RGB-image is transferred into the Lab color space [10]. The L component ist now used to separate the passpoints from the background [11]. In order to further enhance the image quality, morphological operations like *erosion* und *dilatation* are applied [12]. Afterwards each passpoint i with $i = 1, 2, \ldots, 12$ is represented by a cohesive region R_i containing discrete coordinates (n_1, n_2). The centroid C_i of each passpoint can now be determined by

$$C_i = \begin{cases} \frac{1}{N_1} \sum\limits_{n_1} n_1 & \forall n_1 \in R_i \\ \frac{1}{N_2} \sum\limits_{n_2} n_2 & \forall n_2 \in R_i \,. \end{cases} \qquad (3)$$

Due to the meanvalue in Eq. 3, this calculation is performed with inherent subpixel accuracy (results see Figure 4). On the other hand, the measurement of corresponding 3D world coordinates in the pass point frame (see Figure 3) is limited. So using more

than two digits after decimal point is not useful here.
As the passpoints are *located* now, they have to be *identified*. Defined by the passpoint frame, each point got a certain number (see Figure 3 (a)), which has to be assigned correctly and independently from the specific orientation of the frame. Each side (back, bottom, left) of the frame contains four passpoints. So, points in each plane will have the same orientation of their major axis. Based on that, all points are assigned to the corresponding group.
The identification inside one group is now based on the ab color information. All sides contain points of different colors, which can clearly be separated in the Lab color space. To determine the similarity of one color to a given reference, the euklidean distance

$$\Delta = \sqrt{\Delta_L^2 + \Delta_a^2 + \Delta_b^2} \tag{4}$$

is applied. Here, the selection of well-defined colors [8] is both necessary and useful. So for each frame side *independently*, four color "spots" are detected. Based on a look-up table, the identification is finally performed by assigning the corresponding point number to the detected color spot. Table 1 shows the results of this automatic color based identification.

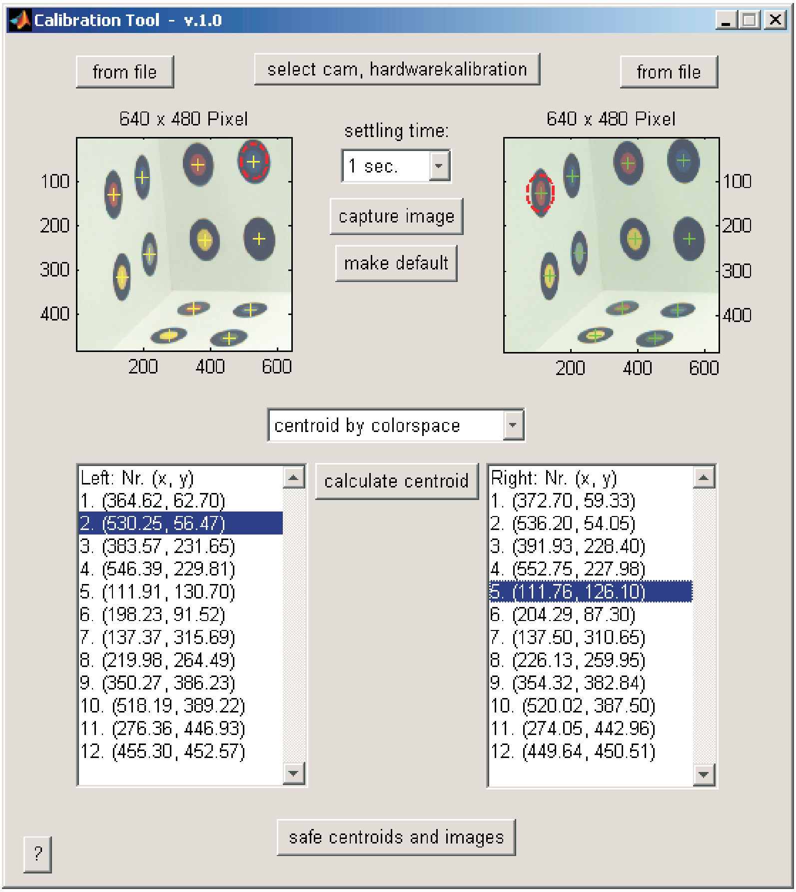
Object	Orientation	Plane	Color	Point No.
a	$8,7$	bottom	red	9
g	$9,2$		yellow	11
b	$9,3$		blue	10
j	$10,1$		green	12
f	$23,7$	back	blue	2
c	$23,9$		black	4
e	$24,8$		red	1
i	$25,0$		yellow	3
k	$68,7$	left	red	5
d	$69,4$		blue	6
h	$69,6$		orange	8
l	$69,8$		yellow	7

Table 1: Assignments between orientation, planes, colors and point numbers.

In the first column the characters denote the order, in which the points were initially detected by the algorithm. Corresponding orientations (in degrees) were obtained using the Matlab-command `regionprops` [13]. With that the assignment to the different planes (bottom, back, left) is performed. The results of the color identification based on Eq. 4 is given next. Finally, the assignment of point numbers complies with the requirements in Figure 3.

4. Results

For demonstration reasons, all functionality has been put into a graphical user interface [9]. Figure 4 shows the results of this approach. In the upper half, the captured images from the stereo camera setup are displayed. Each found centroid is marked by a small yellow cross. The lower half contains two lists with the corresponding image coordinates of the centroids shown above. The actual selected passpoint from the list is emphasized by a red circle in the image.



Left: Nr. (x, y)
1. (364.62, 62.70)
2. (530.25, 56.47)
3. (383.57, 231.65)
4. (546.39, 229.81)
5. (111.91, 130.70)
6. (198.23, 91.52)
7. (137.37, 315.69)
8. (219.98, 264.49)
9. (350.27, 386.23)
10. (518.19, 389.22)
11. (276.36, 446.93)
12. (455.30, 452.57)

Right: Nr. (x, y)
1. (372.70, 59.33)
2. (536.20, 54.05)
3. (391.93, 228.40)
4. (552.75, 227.98)
5. (111.76, 126.10)
6. (204.29, 87.30)
7. (137.50, 310.65)
8. (226.13, 259.95)
9. (354.32, 382.84)
10. (520.02, 387.50)
11. (274.05, 442.96)
12. (449.64, 450.51)

Figure 4: Result for color based passpoint identification.

The results are then forwarded to the calibration and rectification procedure (see Sec. 2). In the rectified image pair, depth estimation or coding can now be performed based on disparity estimation algorithms, e.g. [6]. Figure 5 shows a stereoscopic image pair


(a) Left image (b) Right image

Figure 5: Stereo image pair after calibration and rectification.

after calibration and rectification. The scene contains various objects in different distances from the camera. Between the left and right image, the horizontal displacement (i.e. *disparity*) can be figured out easily when looking at the left hand side of the images, where the emergency switch appears. The aim of stereoscopic calibration and rectification was to ensure standard stereo geometry. So corresponding pixels should be found *in the same row* in both images. Verifying this in the image pair for several characteristic points (e.g. object corners), it is obvious that the goal has been reached very accurate.
With this setup, typical stereoscopic vision tasks are reduced to a one-dimensional problem, reducing complexity and thus saving computational load.

5. Conclusion and Outlook

We have presented an approach for *automatic* identification and localisation of passpoints based on color information. It is part of a stereo camera setup, where the calibration step is necessary in order to rectify the image pair into standard stereo geometry. Here, the identification of passpoints works fully automatical without any user invention.
The image is first transferred into the *Lab* color space. Based on the luminance component, the point centroids can be determined. Orientation is then used to group points in each plane. Finally, the passpoints are identified be separating and detecting different colors. The developped graphical user interface allows a "one-button-press" strategy from capture to rectification.
Future work will deal with the processing of video sequences after the calibration step. With a subsequent interframe disparity estimation process, the change of depth in the scene due to moving objects will become obvious and observable.

References

[1] O. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint.* MIT Press, Cambridge, 1993.

[2] R.Y. Tsai. *An efficient and accurate camera calibratrion technique for 3D machine vision.* Proc. CVPR, Miami Beach, USA, 1986, pp.364–374.

[3] R. Klette, A. Koschans, C. Schlüns. *Computer Vision.* Springer, 1998.

[4] R. Opitz. *Kalibrierung und Rektifikation realer konvergenter Stereoansichten.* Master's Thesis (in german), Helmut-Schmidt-University, Hamburg, Germany, June 2004.

[5] A. Fusiello. *A compact algorithm for Rectification of Stereo Pairs*, Journal of Machine Vision and Applications, Vol. 12, pp. 16-22, 2000.

[6] U. Ahlvers, U. Zölzer. *Improvement of Phase-based Algorithms for Disparity Estimation by Means of Magnitude Information.* In Proc. ICIP, Singapore, 2004.

[7] U. Ahlvers, U. Zölzer, G. Heinrich. *Adaptive Coding, Reconstruction and 3D Visualisation of Stereoscopic Image Data.* In Proc. IASTED VIIP, Marbella, Spain, 2004.

[8] RAL. *Deutsches Institut für Gütesicherung und Kennzeichnung.* `http://www.ral.de`, 2005.

[9] B. Keck. *Automatische Separation und Identifikation von Passpunkten zur Kalibrierung eines Stereokamera-Systems.* Project Work (in german), Helmut-Schmidt-University, Hamburg, Germany, March 2005.

[10] W. Pratt. *Digital Image Processing.* J. Wiley and Sons, Los Altos, California, 3^{rd} Edition, 2001.

[11] N. Otsu. *A threshold selection method from gray level histograms.* IEEE Trans. Systems, Man and Cybernetics, Vol.9, pp.62–66, 1979.

[12] R. Klette and P. Zamperoni. *Handbook of Image Processing Operators.* J. Wiley and Sons, 1^{st} Edition, 1996.

[13] The Mathworks. Image Aquisition Toolbox. `http://www.mathworks.com/products/imaq/`, 2003.