EXTRINSIC CALIBRATION OF A STEREO CAMERA SYSTEM USING A 3D CAD MODEL CONSIDERING THE UNCERTAINTIES OF ESTIMATED FEATURE POINTS

Kai Cordes, Patrick Mikulastik, Alexander Vais, Jörn Ostermann

Institut für Informationsverarbeitung (TNT)
Hannover, Germany
http://www.tnt.uni-hannover.de

Abstract

The extrinsic calibration of a stereo camera system is a procedure that estimates the position and orientation of a stereo camera system relative to a calibration object. In this paper, a calibration object specified by a CAD model of known shape and arbitrary texture is used. Based on feature points located in the stereo image and using triangulation, 3D object points in the camera coordinate system are calculated. This point cloud is matched to the surface of the CAD model to estimate the position and the rotation of the stereo camera system relative to the object coordinate system. In order to improve the accuracy of known techniques for the calibration, two steps are proposed in this paper. First, a systematic error in the standard method of subpel feature point localization is eliminated by replacing the parabolic interpolation by a Gaussian regression. Second, the estimated accuracies of the 3D points are incorporated by propagating the uncertainties of the detected feature points in the images. To calculate the global minimum of the cost function, an evolutionary optimizer is combined with a two-stage strategy, refining a fast converging approximate result. The developed procedure reduces the error variance of observed points by a factor of about 80.

Keywords: stereo vision, camera calibration, reconstruction, registration, uncertainty propagation

1 Introduction

The extrinsic calibration of a stereo camera system can be done using a 3D object of known shape and arbitrary texture. Hence, the calibration is the task of matching the object coordinate system of the 3D CAD model to the local coordinate system of the stereo camera. For a stereo camera system with known intrinsic parameters and known baseline, this problem can be solved by finding the six parameters of a Euclidean transformation [7, 21]. These parameters are obtained by performing the following procedures. First, corresponding feature points are detected in the stereo image capturing the calibration object. Then, 3D coordinates of the feature points are calculated using triangulation. Finally, the Euclidean transformation minimizing the distances between the 3D points and the surface of the CAD model is estimated. This transformation represents the mapping between the camera coordinate system and the object coordinate system. The matching of a point cloud to a 3D CAD model is known as registration [16, 17].

The Canny detector [2] is still one of the most popular techniques for detecting edges in an image with moderate computational expense. To estimate the position of an edge with subpel accuracy, Rockett [14] proposed an extension to Canny's approach. The discrete image signal gradients are interpolated using a parabolic fitting curve to increase the accuracy. However, it was shown [4, 11, 14] that using a parabolic approximation of image signal gradients still leads to a systematic error in the subpel estimation.

The registration of a point cloud to a 3D model is usually done using the *Iterative Closest Point* (ICP) approach [15]. It maps each 3D feature point to a point on the 3D CAD model while minimizing the distances iteratively. The problem of the dependency of the results on the initial parameter configuration was solved in [17] using the *Differential Evolution* (DE) algorithm for the optimization. DE is an efficient evolutionary algorithm, that is capable of solving continuous global optimization problems [19]. Furthermore, a robust cost function was introduced in [17] to be able to deal with outliers. Outliers are 3D points that have a large distance to the 3D CAD model caused by inaccuracies in the feature point detection or mismatches in the correspondence analysis.

In our work, the error in the subpel feature point localization is reduced by replacing the parabolic interpolation by a Gaussian regression. The remaining uncertainty of a feature point is derived from the surrounding image gradients and the estimated Gaussian regression function. The uncertainty is propagated using the mapping equations of the stereo camera. Furthermore, the cost function is extented by means of the Mahalanobis distance [9] to be able to incorporate the uncertainties of triangulated feature points. Uncertainty propagation has been found to be an appropriate method to increase the accuracy of 3D reconstruction approaches [3, 18].

In the following Section the calibration procedure is briefly presented. In Section 3, the detection of feature points and the

978-0-7695-3893-8/09 $26.00 © 2009 IEEE
DOI 10.1109/CVMP.2009.10

usage of their uncertainties in the estimation is derived. Section 4 shows the optimization strategy. In Section 5, results of the experiments and convergence properties of the optimization procedure are shown, and in Section 6 the paper is concluded.

2 Calibration of a Stereo Camera System using a Calibration Object

For the extrinsic calibration of the stereo camera using a calibration object we need to find the mapping between the object coordinate system of the CAD model determining the calibration object and the camera coordinate system of the stereo camera. Our input data is a stereo image pair capturing the calibration object. As shown in Figure 1, we assume a standard stereo camera configuration. This means, that both equally oriented camera targets share the same plane. The calibration object has known shape specified by the CAD model. First, edges are detected in the images using the Canny detector [2]. For the estimation of the subpel x-coordinate of a feature point p_c, a Gaussian approximation of the image signal gradients is used (see Section 3.3). For the y-coordinate, the fullpel position is assumed. Following from the standard stereo approach, the y-coordinate is equal in both images ($y_{cl} = y_{cr}$). Feature correspondences are established using blockwise normalized cross correlation between a feature point in the left and all candidates inside a search area in the right camera image. As we are assuming a standard stereo camera setup, the corresponding feature point can be found in the same image scanline. Then, a 3D point P_C is calculated using the disparity $x_{cl} - x_{cr}$ of the corresponding points p_{cl} and p_{cr} in the left and right camera image (see Figure 1). The depth Z_C of P_C captured by a stereo camera with the baseline $|\vec{X}_B|$ and the focal length f is calculated as:

$$Z_C = \frac{f \cdot |\vec{X}_B|}{x_{cl} - x_{cr}} \tag{1}$$

Assuming a camera coordinate system with its origin at the position of the left camera (see Figure 1), the vector $\vec{P}_C$ pointing to P_C can be written as $\vec{P}_C = (X_C, Y_C, Z_C)^\top$. The components X_C and Y_C follow by means of the detected point $p_{cl} = (x_{cl}, y_{cl})$ in the left image:

$$\begin{pmatrix} X_C \\ Y_C \end{pmatrix} = \frac{Z_c}{f}\vec{p}_{cl} = \frac{Z_c}{f}\begin{pmatrix} x_{cl} \\ y_{cl} \end{pmatrix} \tag{2}$$

The point P_C described in the camera coordinate system has to be transformed to a point P_O in the object coordinate system through a Euclidean transformation T. The point P_O represents a position on the calibration object described by the 3D CAD model. The mapping T is:

$$T : \vec{P}_O = \mathbf{R} \cdot \vec{P}_C + \vec{C} \tag{3}$$

where $\mathbf{R}$ is a 3 x 3 rotation matrix and $\vec{C} = (C_x, C_y, C_z)^\top$ is the position of the left camera in the object coordinate system. Thus, this mapping can be described by three angles ϑ, φ, and ϱ

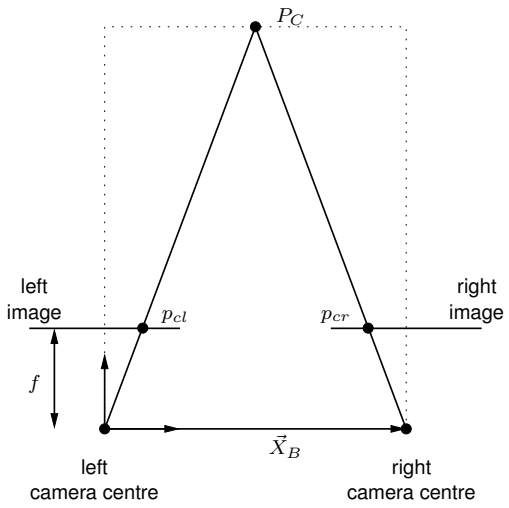

Figure 1: *Projection of a 3D point P_C into the left and right image of the stereo camera*

and three position components C_x, C_y, and C_z. To obtain these parameters, the Differential Evolution (DE) algorithm is used. DE is a global optimization technique, that minimizes a cost function while varying its parameters iteratively (see Section 4). The cost function to be minimized is the sum of squared distances $(d_{P_C})^2$ between each point P_C of the point cloud and the surface of the 3D CAD object [20]. In order to be able to deal with outliers, a robust cost function is used as explained in the next section.

2.1 Robust Cost Function

The cost function that has to be minimized, adds up all squared distances $(d_{P_C})^2$ from each of N points of the point cloud to the CAD model of the calibration object [20]. This is done as proposed in [17]:

$$D_r(\mathbf{R}, \vec{C}) = \sum_{n=0}^{N-1} \begin{cases} \frac{1}{\kappa_r} \cdot (d^n_{P_C})^2 & \text{, for } (d^n_{P_C})^2 \leq \kappa_r \\ 1 & \text{, otherwise} \end{cases} \tag{4}$$

Here, κ_r is a scaling factor that controls the influence of a point P_C on the cost function. If the distance of P_C is larger than a threshold ($(d_{P_C})^2 > \kappa_r$) its influence does not increase with the distance, it stays constant. A point with a distance greater than this threshold is considered an outlier. In [17], $\kappa_r = 5$ and the Euclidean distance measure are used.

3 Evaluation of the Feature Point Position

Assuming a standard stereo setup with corresponding feature points having the same y-coordinate, for the following only the x-coordinate with the fullpel value $n_x \in \mathbb{Z}$ and the subpel value $x_0 \in [-0.5; 0.5]$ of an image point will be considered. To calculate the gradient of the image signal, the following filter

is used [2, 8]:

$$I_{grad}(n_x) = \sum_{i=-2}^{2} I(n_x - i) \cdot \frac{-i}{\sqrt{2\pi}\sigma_{grad}^3} e^{\frac{-i^2}{2\sigma_{grad}^2}} \quad (5)$$
$$\text{, with } \sigma_{grad} = 1.0$$

In [11] it is shown, that using a parabolic interpolation function for the subpel interpolation of detected feature points leads to a systematic error of 0.025 pel at positions with a subpixel coordinate of ± 0.25 pel. This systematic error can be avoided by replacing the parabolic interpolation by a Gaussian regression [4, 10], as the pulse response of an image acquisition system can be modelled by a Gaussian [1, 5]. Hence, for the estimation of the subpel coordinate of a feature point, a Gaussian function is fitted to the discrete gradient values in the horizontal image scanline around an edge. The accuracy of this coordinate determines the estimated accuracy of the 3D point position obtained by triangulation. In this paper, five grid points with the x-coordinates $n_x - 2, n_x - 1, n_x, n_x + 1, n_x + 2$ are used for estimating the Gaussian regression function.

3.1 Approximation Error of the Gaussian

The regression function used for the estimation of the subpel x-coordinate x_0 of a feature point is the following Gaussian with mean value x_0, variance σ_g, and a scaling factor K:

$$G(x, \sigma_g, x_o, K) = K \cdot \frac{1}{\sqrt{2\pi}\sigma_g} e^{-\frac{1}{2}(\frac{x - x_0}{\sigma_g})^2} \quad (6)$$

The estimated solution parameter vector $\tilde{\vec{v}}$ for the Gaussian function follows through regression analysis:

$$\tilde{\vec{v}} = \begin{pmatrix} \hat{\sigma}_g \\ \hat{x}_0 \\ \hat{K} \end{pmatrix} \quad (7)$$

This vector $\tilde{\vec{v}}$ can be written as the sum of the exact solution $\vec{v}_{opt}$ and the approximation error $\vec{\epsilon}_v$:

$$\tilde{\vec{v}} = \vec{v}_{opt} + \vec{\epsilon}_v \quad (8)$$

The covariance matrix $\mathbf{C}_{\epsilon_v}$ of $\vec{\epsilon}_v$ leads to the error variance $\sigma_{\epsilon_x}^2$ of a feature point position in x-direction (Section 3.3). This $\mathbf{C}_{\epsilon_v}$ can be used to calculate the uncertainty of the corresponding triangulated 3D feature point (Section 3.4).

3.2 Distortion of the Image Signal Gradient

In the following, the camera noise is modelled by the function r with a variance σ_r^2 at position $n_x \in \mathbb{Z}$:

$$\tilde{I}(n_x) = I(n_x) + r(n_x) \quad (9)$$

Using a five tap Gaussian regression function, the distorted image signal gradient $\tilde{I}_{grad}(n_x)$ can be written as (see (5)):

$$\begin{aligned} \tilde{I}_{grad}(n_x) &= \sum_{i=-2}^{2} \tilde{I}(n_x - i) \cdot \tfrac{-i}{\sqrt{2\pi}} e^{\frac{-i^2}{2}} \\ &= \sum_{i=-2}^{2} (I(n_x - i) + r(n_x - i)) \tfrac{-i}{\sqrt{2\pi}} e^{\frac{-i^2}{2}} \\ &= I_{grad}(n_x) + \sum_{i=-2}^{2} r(n_x - i) \tfrac{-i}{\sqrt{2\pi}} e^{\frac{-i^2}{2}} \end{aligned} \quad (10)$$

So, the distortion $\epsilon_{grad}(n_x)$ of the image signal gradient is:

$$\begin{aligned} \epsilon_{grad}(n_x) &= \tilde{I}_{grad}(n_x) - I_{grad}(n_x) \\ &= \sum_{i=-2}^{2} r(n_x - i) \cdot \frac{-i}{\sqrt{2\pi}} e^{\frac{-i^2}{2}} \end{aligned} \quad (11)$$

For calculating the covariance matrix

$$\mathbf{C}_{\epsilon_{grad}} = E\left[\vec{\epsilon}_{grad} \vec{\epsilon}_{grad}^{\top}\right]$$

$$= E\left[\begin{pmatrix} \epsilon_{grad}(-2) \\ \vdots \\ \epsilon_{grad}(+2) \end{pmatrix} \cdot (\epsilon_{grad}(-2), \ldots, \epsilon_{grad}(+2)) \right] \quad (12)$$

we assume statistically independent noise:

$$E\left[r(n_i) \cdot r(n_j)\right] = \begin{cases} 0 & \text{, for } n_i \neq n_j \\ \sigma_r^2 & \text{, for } n_i = n_j \end{cases} \quad (13)$$

Here, σ_r^2 is the a-priori known camera noise variance. Now, the symmetric 5×5-matrix $\mathbf{C}_{\epsilon_{grad}}$ can be calculated using (11), (12), and (13).

3.3 Uncertainty of the Position of a Feature Point

The error $\vec{\epsilon}_v$ in (8) can be described by the regression function G and the distortion of the image signal gradient $\vec{\epsilon}_{grad}$. Evaluating the error function $e(\vec{I}_{grad}, \vec{G})$,

$$e(\vec{I}_{grad}, \vec{G}) = \left| \vec{I}_{grad} - \vec{G} \right|^2 \quad (14)$$

of the gradient grid points $\vec{I}_{grad} = (I_{grad}(-2), \ldots, I_{grad}(2))^{\top}$ and the values of the regression function $\vec{G} = (G(-2, \vec{v}^{\top}), \ldots, G(2, \vec{v}^{\top}))^{\top}$ at its minumum $\vec{v} = \vec{v}_{opt}$,

$$\nabla_{\vec{v}} [e(\vec{I}_{grad}, \vec{G})] \Big|_{\vec{v} = \vec{v}_{opt}} = \vec{0}$$

leads to the following formula [10]:

$$\vec{\epsilon}_v = \underbrace{\left(\left[\left[\nabla_{\vec{v}} \vec{G}^\top \right] \left[\nabla_{\vec{v}} \vec{G}^\top \right]^\top \right]^\top \right)^{-1} \cdot \nabla_{\vec{v}} \vec{G}^\top \Big|_{\vec{v}=\vec{v}_{opt}}}_{=:\mathbf{B}} \cdot \vec{\epsilon}_{grad}$$
$$= \mathbf{B} \cdot \vec{\epsilon}_{grad} \quad (15)$$

Here, $\nabla_{\vec{v}} = (\frac{\partial}{\partial \sigma_g}, \frac{\partial}{\partial x_0}, \frac{\partial}{\partial K})^\top$ denotes the partial derivatives of the parameter vector $\vec{v}$, and $\vec{\epsilon}_{grad} = (\epsilon_{grad}(-2), \ldots, \epsilon_{grad}(+2))^\top$ can be approximated using (11).

To obtain the error variance of the feature point position $\sigma^2_{\epsilon_x}$, the covariance matrix $\mathbf{C}_{\epsilon_v}$ is calculated using (15) and (12) as follows:

$$\mathbf{C}_{\epsilon_v} = E\left[\vec{\epsilon}_v \vec{\epsilon}_v^\top\right] = \mathbf{B} \cdot E\left[\vec{\epsilon}_{grad} \vec{\epsilon}_{grad}^\top\right] \cdot \mathbf{B}^\top$$
$$= \mathbf{B} \cdot \mathbf{C}_{\epsilon_{grad}} \cdot \mathbf{B}^\top \quad (16)$$

with

$$\mathbf{C}_{\epsilon_v} = \begin{bmatrix} \sigma^2_{\epsilon_\sigma} & \cdots & \\ \vdots & \sigma^2_{\epsilon_x} & \vdots \\ & \cdots & \sigma^2_{\epsilon_K} \end{bmatrix} \quad (17)$$

Here, $\sigma^2_{\epsilon_x}$ is the variance of the position error of the feature point in x-direction. The value $\sigma^2_{\epsilon_x}$ leads to the estimated uncertainty of a 3D point as explained in the next section.

3.4 Error Approximation of a 3D Point Position

Using (1) and (2) we can write the vector $\vec{P}_C$ to a 3D point P_C by means of the detected corresponding feature points (x_l, y_l) and (x_r, y_r) with $y_l = y_r$ (following from standard stereo) as:

$$\vec{P}_C = \frac{|\vec{X}_B|}{x_l - x_r} \begin{pmatrix} x_l \\ y_l \\ f \end{pmatrix} \quad (18)$$

For small errors $(\epsilon_{x_l}, \epsilon_{x_r})^\top$, we can approximate the erroneous 3D position of the calculated point

$$\vec{P}_C(x_l + \epsilon_{x_l}, x_r + \epsilon_{x_r}) = \vec{P}_C(x_l, x_r) + \mathbf{J}_{P_C} \begin{pmatrix} \epsilon_{x_l} \\ \epsilon_{x_r} \end{pmatrix} \quad (19)$$

with the help of the Jacobian matrix $\mathbf{J}_{P_C}$ consisting of partial derivatives $\frac{\partial \vec{P}_C}{\partial (x_l, x_r)}$:

$$\mathbf{J}_{P_C} = \left(\frac{\partial \vec{P}_C}{\partial (x_l, x_r)} \right) = \frac{|\vec{X}_B|}{(x_l - x_r)^2} \begin{pmatrix} -x_r & x_l \\ -y_l & y_l \\ -f & f \end{pmatrix} \quad (20)$$

Assuming an equal error variance $\sigma^2_{\epsilon_x}$ in corresponding feature point positions in the left and right image, we can formulate the covariance matrix of each 3D point P_C:

$$\mathbf{C}_{\epsilon P_C} = \mathbf{J}_{P_C} \mathbf{J}_{P_C}^\top \sigma^2_{\epsilon_x} \quad (21)$$

Performing the matrix multiplication (21) and substituting $(x_l - x_r)$ with the help of (1), it follows:

$$\mathbf{C}_{\epsilon P_C} = \mathbf{J}_{P_C} \mathbf{J}_{P_C}^\top \sigma^2_{\epsilon_x}$$
$$= \frac{Z_C^4 \sigma^2_{\epsilon_x}}{f^2 |\vec{X}_B|^2} \begin{pmatrix} \frac{x_l^2 + x_r^2}{f^2} & \frac{y_l (x_l + x_r)}{f^2} & \frac{x_l + x_r}{f} \\ \frac{y_l (x_l + x_r)}{f^2} & \frac{2 y_l^2}{f^2} & \frac{2 y_l}{f} \\ \frac{x_l + x_r}{f} & \frac{2 y_l}{f} & 2 \end{pmatrix} \quad (22)$$

Notice, that the three diagonal matrix elements describe the error variance in the x- , y- , and z - direction of the camera coordinate system. For our experiments, we use a stereo camera with a focal length of $f = 30.072$ mm, $|x_l|, |x_r| < 3.437$ mm, and $|y_l| < 1.933$ mm. So, the covariances of P_C in x- and y- direction are much smaller than the covariance in z-direction. A visualisation of the uncertainty of P_C is shown in Figure 2.

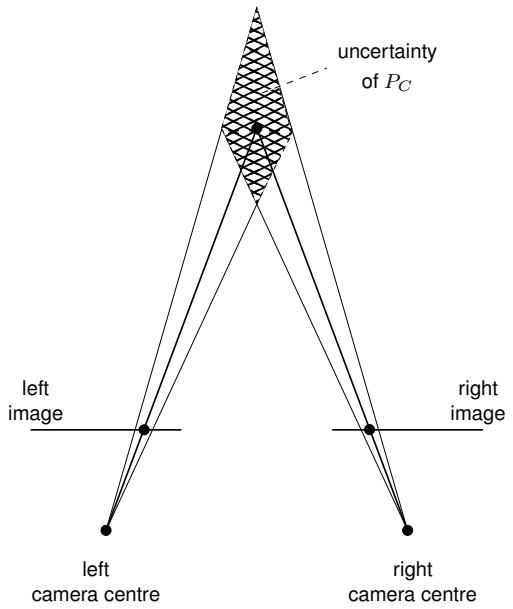

Figure 2: *Uncertainty of the 3D point P_C resulting from the distortion of the x-coordinates of the feature point positions in the stereo image*

3.5 Incorporating Uncertainty Information

In order to incorporate the covariance information $\mathbf{C}_{\epsilon P_C}$ for each point P_C in our cost function, the Mahalanobis distance

measure ξ_n [6, 9] in equation (4) is used:

$$D_m(\mathbf{R},\vec{C}) = \sum_{n=0}^{N-1} \begin{cases} \frac{1}{\kappa_m} \cdot (\xi_{P_C}^n)^2 & \text{, for } (\xi_{P_C}^n)^2 \leq \kappa_m \\ 1 & \text{, otherwise} \end{cases} \tag{23}$$

An appropriate value for κ_m is found by means of an experiment and is set to $\kappa_m = 16.3$ for the experiments in this paper. In contrast to the Euclidean distance measure, the minimal Mahalanobis distance vector between a 3D point and the object's surface is usually not orthogonal to the surface. Points of the same Mahalanobis distance from a fixed 3D point lie on an ellipsoid with maximum extent in the direction of the biggest uncertainty as shown in Figure 3. The ellipsoid is defined in terms of the estimated uncertainties in x-, y-, and z-direction in (22).

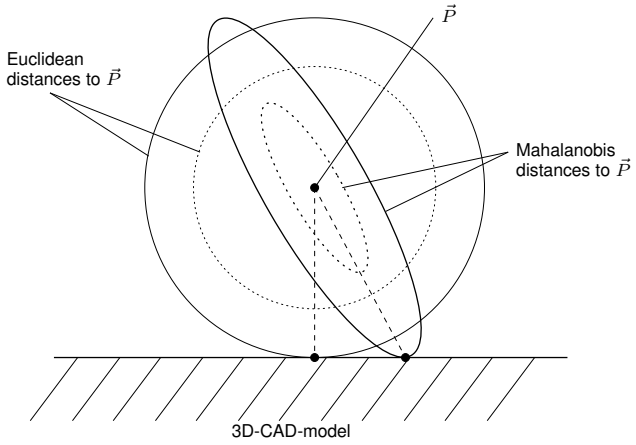

Figure 3: *Comparison of the Mahalanobis distance ξ and the Euclidean distance between a point $\vec{P}$ and the surface of a 3D-CAD-model*

4 Differential Evolution Optimization

For optimizing the cost function (23), the Differential Evolution (DE) algorithm [13, 19] is used. It is known as an efficient global optimization method for continuous problem spaces. The optimization is based on a population of Np solution candidates $x_{n,i}$, $n \in [1; Np]$, at iteration i where each candidate is a position in the 6-dimensional search space. Initially, the solution candidates are randomly generated within the provided intervals of the search space. The population improves by generating new 6-tupels iteratively for each candidate. New positions for the iteration step $i+1$ are determined by

$$y_{n,i+1} = x_{k,i} + F \cdot (x_{l,i} - x_{m,i}) \tag{24}$$

$$x_{n,i+1} = c\,(x_{n,i},\ y_{n,i+1}) \tag{25}$$

where k, l, m are random integers from the interval $[1; Np]$, $F \in [0; 1]$ is a weighting scalar, $y_{n,i+1}$ is a displaced $x_{k,i}$ by a weighted difference vector, and $c\,(x_{n,i},\ y_{n,i+1})$ is a crossover operator copying coordinates from both $x_{n,i}$ and $y_{n,i+1}$ in order to create $x_{n,i+1}$. The crossover operator c is provided with a value $Cr \in [0; 1]$ specifying the probability to copy values either from $x_{n,i}$ or $y_{n,i+1}$ to $x_{n,i+1}$ for each component of the vector. Only if the new candidate $x_{n,i+1}$ proves to have lower cost using a cost function it replaces $x_{n,i}$, otherwise it is discarded.

DE includes an adaptive range scaling for the generation of solution candidates through the difference term in (24). This enables global search in the case where the solution candidate vectors are spread in the search space and the mean difference vector is relatively large. In the case of a converging population the mean difference vector becomes relatively small, and this enables efficient fine tuning at the end phase of the optimization process [20].

4.1 Two-stage Algorithm

Some tests with our data reveal different convergence behaviour of DE for our application using different cost functions. While using simple Euclidean distances in the cost function D_e,

$$D_e(\mathbf{R},\vec{C}) = \sum_{n=0}^{N-1} (d_{P_C}^n)^2 \tag{26}$$

provides fast convergence, the robust cost functions D_r (4) and D_m (23) lead to convergence problems, because of the optimization algorithm getting stuck in a local minimum of the cost function. The convergence behaviour of each cost function presented here is discussed in Section 5.1 in detail.

In literature [16], two-stage solutions are proposed for ICP, calculating an approximation in a first step and obtaining the accurate result in a second refinement step. This idea can be transfered to the DE approach by minimizing the cost function D_e first to get quickly close to the global optimum. In a second step, the cost function D_m is minimized, using the resulting population of the first step as initial population.

The convergence behaviour of this two-stage strategy compared to the classical one-stage DE is also shown in Section 5.1.

5 Experimental Results

For our experiments, we use synthetic and natural image data. The synthetic images are rendered using PovRay [12] with a resolution of 640x480 pel. The synthetic stereo camera captures the calibration object shown in Figure 4. An image pair can be seen in Figure 5.

Natural images (Figure 6) are captured using two calibrated Thomson Viper CCD-cameras with a resolution of 1920x1080 pel and a-priori known intrinsic camera parameters. The stereo baseline is $|\vec{X}_B| = 250.33$ mm. The distance to the object is about 2 m. The natural images are rectified and radial distortion is compensated to guarantee the standard stereo assumption. The stereo camera captures a real calibration object with the shape of the CAD model in Figure 4.

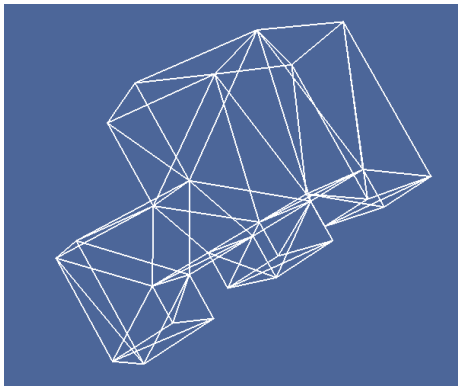

Figure 4: *Triangulation of the calibration object*

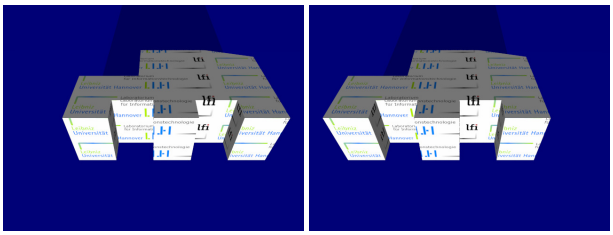

Figure 5: *Left and right synthetic image of a virtual calibration object*

For the synthetic images, white Gaussian noise of different amount is added to the rendered images. Three different noise levels are used: 38.1 dB, 33.4 dB, and 31.1 dB. The mean noise level of the real camera is 38.1 dB.

For the optimization with the DE algorithm, we use common parameters [13] and choose the weighting scalar $F = 0.8$ and the crossover probability $Cr = 0.8$. The number of particles is set to $Np = 100$. In Section 5.1 we discuss the observed convergence behaviour of the common one-stage algorithm and the two-stage strategy developed in this paper. In Section 5.2 the accuracy of the calibration results is presented.

5.1 Convergence Properties

The following results regarding the convergence properties are obtained using the synthetic image pair in Figure 5 with added

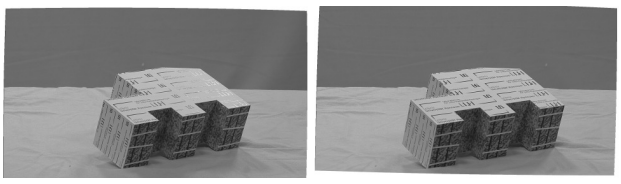

Figure 6: *Rectified left and right camera image of the real calibration object*

	Number of iterations				
d	300	500	1000	2000	4000
D_e	90.2%	100%	100%	100%	100%
D_r	0.0%	0.0%	46.4%	83.8%	83.8%
D_m	0.0%	0.0%	0.0%	3.0%	41.0%
two-stage	0.0%	92.4%	100%	100%	100%

Table 1: *Convergence rate of the Differential Evolution (DE) algorithm using different cost functions d: D_e = Euclidean not robust, D_r = Euclidean robust, D_m = Mahalanobis robust; the proposed two-stage strategy providing the calibration accuracy of D_m.*

Gaussian noise of 38.1 dB. For all other scenes presented in this paper we got similar convergence results.

Using a robust cost function and the Mahalanobis distance affects the convergence behaviour of the global optimization algorithm. A comparison of the convergence rates using different cost functions is shown in Table 1. Therefore, for each method the global minimum of the cost function D^{MIN} is determined. Then, 500 runs with random initializations are evaluated, and the number of runs converging successfully is counted. The convergence is deemed successful if the cost function value is lower than $1.001 \cdot D^{MIN}$.

With an increasing number of iterations of DE, the success rate increases for all methods as expected. In comparison to the non-robust Euclidean distance cost function D_e (26), the robust Euclidean distance D_r (4) needs more iterations to succeed. Furthermore, D_r does not ensure reaching the global minimum, as the identical rates after 2000 and 4000 iterations indicate, that there will not be more success after more than 4000 iterations. The Mahalanobis distance together with the robust cost function D_m (23) leads to even more iterations needed and higher uncertainty for achieving the global minimum. It results in a success rate of only 41 % after 4000 iterations.

The influence of the robust cost function can be explained by a major part of the search space being covered with constant cost function values. The Mahalanobis distance measure narrows the peak of the global optimum additionally. Note, that in our experiments the covariance in z- direction is much higher than in x- and y- directions (see Section 3.4). We assume that these are the reasons for the decreasing convergence rate of the cost functions D_r and D_m compared to the cost function D_e.

The proposed two-stage strategy explained in Section 4.1 provides the calibration accuracy of D_m and a good convergence rate. It uses the Euclidean distance measure D_e in the first stage until a switching criterion is achieved. In this experiment, the switching criterion is a simple threshold value for the cost function D_e, as the minimum D_e^{MIN} of the cost function is known. The threshold is set to $1.001 \cdot D_e^{MIN}$ as explained above, but there are several other criteria possible that do not need an absolute threshold [22]. For the two-stage

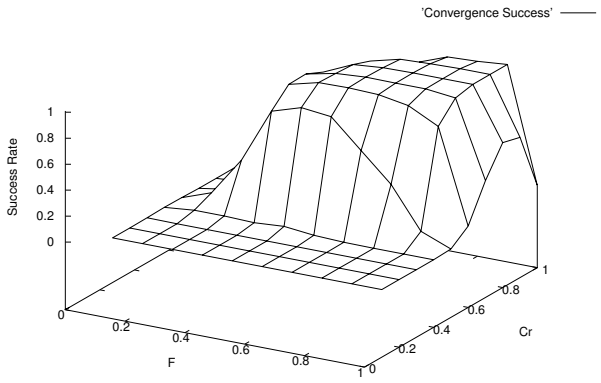

Figure 7: *Convergence success rate of the global optimization for different combinations of the parameters F and Cr of the Differential Evolution (DE) algorithm. Here, the proposed two-stage strategy is used.*

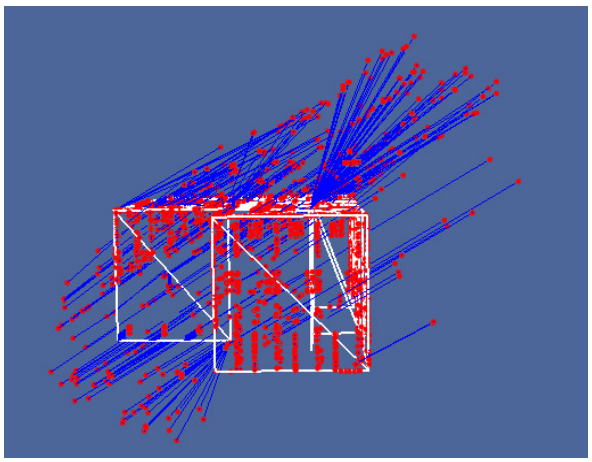

Figure 8: *Side view of the calibration object (Figure 4) and the resulting point cloud using the image pair shown in Figure 6. The blue lines indicate the Mahalanobis distance used for the calibration procedure.*

strategy, 100 % optimization success is achieved after 1000 iterations for both stages together (see Table 1, last row). The two-stage concept needs a few iterations more than D_e, as the strategy has to switch to a different cost function D_m in the beginning of the second stage leading to more accurate results (see Section 5.2).

In Figure 7 convergence results for the proposed strategy are shown for all possible parameter combinations $0.1 \leq F, Cr \leq 1$ of DE, displaying the convergence rate of 500 runs. The graph depicts the typical convergence behaviour of DE as known from literature [13, 23], performing best and providing convergence for the parameters $0.7 < F, Cr < 1$.

5.2 Calibration Results

To evaluate the resulting parameters of the calibration, for each setup - synthetic and natural image data - a sequence of 1000 images with constant scene content is used.

We compare the results of the calibration procedure using a parabolic interpolation function for the feature point localization and the robust Euclidean distance (cost function D_r), called *scheme [A]*, with the results of our proposal using a Gaussian regression function for the localization of the feature points and considering uncertainties of the 3D feature points (cost function D_m), called *scheme [C]*. To evaluate the contributions of the two proposals, in *scheme [B]* only the Gaussian regression function for the localization is used, and no covariance information of the 3D points.

For our experiments, we can assume an unbiased estimator. This means, that the mean value of the calculated parameters of the 1000 images are identical in all schemes [A], [B], and [C]. So, the quality of the results can be compared by means of the error variance $(\sigma^2_{\epsilon_\theta}, \sigma^2_{\epsilon_\phi}, \sigma^2_{\epsilon_\rho}, \sigma^2_{\epsilon_{Cx}}, \sigma^2_{\epsilon_{Cy}}, \sigma^2_{\epsilon_{Cz}})$ of the six parameters of the Euclidean transformation. In order to get a value for the maximum error variance, the error variances of the angles $(\sigma^2_{\epsilon_\theta}, \sigma^2_{\epsilon_\phi}, \sigma^2_{\epsilon_\rho})$ are converted to position errors in x- , y- , and z- direction. Therefore, a point nearby the optical axis is assumed, having a distance of 2 m from the stereo camera. The resulting error vector is added to the position error variance $(\sigma^2_{\epsilon_{Cx}}, \sigma^2_{\epsilon_{Cy}}, \sigma^2_{\epsilon_{Cz}})$, and the maximum σ^2_{max} of the three values is determined.

Im Figure 8, a resulting point cloud using the stereo image from Figure 6 is shown in red. The blue lines indicate the distances to the surface of the calibration object. It can be seen that the distances used for the calibration are strongly influenced by the optical axis of the stereo camera.

5.2.1 Natural Scene

The error variance results and the improvement in the value σ^2_{max}, combining all six parameters for the natural scene are shown in Table 2. In comparison to scheme [A], scheme [C] provides an improvement of a factor of about 80. This means that the position accuracy σ_{max} has improved by a factor of about 9 from 2.28 mm to 0.25 mm.

In Figure 9, the results for all three schemes [A], [B], and [C] are compared. This result shows, that the main contribution to the improvement of the calibration accuracy is performed by the usage of the knowledge of the uncertainties of the 3D points.

5.2.2 Synthetic Scene

The results for σ^2_{max} for the synthetic scene using different noise levels are shown in Figure 10. The error variance for the

	[A]	[C]
$\sigma^2_{\epsilon_\theta}[DEG^2]$	$4.01 \cdot 10^{-3}$	$0.0658 \cdot 10^{-3}$
$\sigma^2_{\epsilon_\phi}[DEG^2]$	$1.79 \cdot 10^{-3}$	$0.0486 \cdot 10^{-3}$
$\sigma^2_{\epsilon_\rho}[DEG^2]$	$11.3 \cdot 10^{-3}$	$0.211 \cdot 10^{-3}$
$\sigma^2_{\epsilon_{Cx}}[mm^2]$	3.21	0.0242
$\sigma^2_{\epsilon_{Cy}}[mm^2]$	2.63	0.00892
$\sigma^2_{\epsilon_{Cz}}[mm^2]$	1.27	0.0158
$\sigma^2_{max}[mm^2]$	5.18	0.063

Table 2: *Error variance results of the six parameters of the calibration of schemes [A] and [C] using 1000 natural image pairs.*

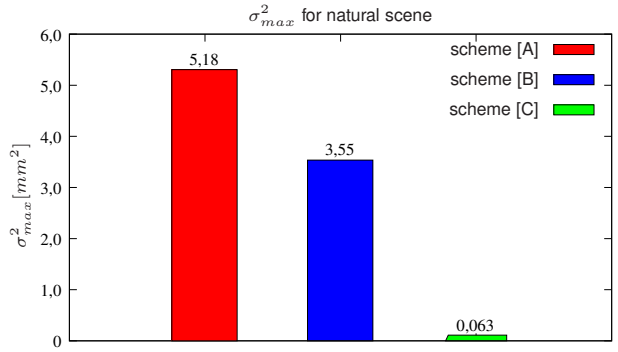

Figure 9: *Error variance σ^2_{max} for 1000 natural image pairs with a noise level of 38.1 dB using schemes [A], [B], and [C]*

noise level of 38.1 dB is matching the result for the natural image data (Figure 9). With an increasing noise level, our proposal [C] yields a nearly constant error variance while the error of the reference [A] increases.

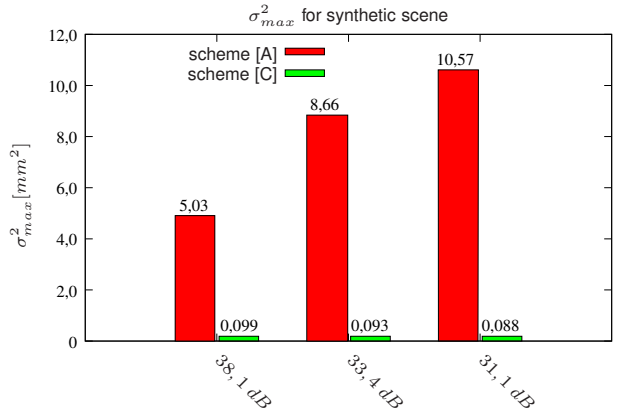

Figure 10: *Error variance σ^2_{max} for 1000 synthetic image pairs with different noise levels using schemes [A] and [C]*

6 Conclusions

In this paper, two techniques are proposed for improving the accuracy of the extrinsic calibration of a stereo camera system using a calibration object of known shape. The first technique employs a Gaussian fitting curve for the subpel estimation of feature points. The second is the incorporation of the estimated accuracies of triangulated 3D points into the registration algorithm. In order to get 100 % convergence success, a two-stage optimization strategy is developed. It uses the solution of a Euclidean cost function as a starting point for the optimization with Mahalanobis distances between the 3D points and the calibration object.

The accuracy of the calibration is improved by a factor of 80 in the error variance. This means, that the position error of a point is improved by a factor of 9 from 2.28 mm to 0.25 mm in a setup with a stereo camera having a distance of about 2 m from the calibration object.

References

[1] K. Aström and A. Heyden. Stochastic modelling and analysis of sub-pixel edge detection. In *IEEE International Conference on Pattern Recognition (ICPR)*, pages 86–90, Vienna, Austria, 1996.

[2] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 8(6):679–698, 1986.

[3] A. R. Chowdhury and R. Chellappa. Statistical error propagation in 3d modeling from monocular video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshop*, volume 8, Madison, USA, 2003.

[4] K. Cordes, O. Müller, B. Rosenhahn, and J. Ostermann. Half-sift: High-accurate localized features for sift. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshop*, Miami Beach, USA, 2009.

[5] J. W. Goodman. *Introduction to Fourier Optics*. McGraw-Hill, 1968.

[6] R. I. Hartley and A. Zisserman. *Multiple View Geometry*. Cambridge University Press, UK, 2000.

[7] R. Horaud and F. Dornaika. Hand-eye calibration. *International Journal of Robotics Research (IJRR)*, 14(3):195–210, 1995.

[8] Y. Ma, S. Soatto, J. Kesecká, and S. Sastry. *An Invitation to 3-D Vision, From Images to Geometric Models*. Springer, New York, USA, 2004.

[9] P. C. Mahalanobis. On the generalised distance in statistics. In *Proceedings of the National Institute of Science*, number 1, pages 49–55, India, 1936.

[10] P. Mikulastik. *Verbesserung der Genauigkeit der Selbstkalibrierung einer Stereokamera mit 3D-CAD-Modellen*. VDI, Germany, 2009.

[11] P. Mikulastik, H. Broszio, T. Thormählen, and O. Urfalıoğlu. Error analysis of feature based disparity estimation. In *Pacific-Rim Symposium on Image and Video Technology (PSIVT)*, volume 4319, pages 1–12, Taiwan, 2006. Springer, Germany.

[12] Povray. Povray - the persistence of vision raytracer, 2007.

[13] K. V. Price, R. Storn, and J. A. Lampinen. *Differential Evolution - A Practical Approach to Global Optimization*. Natural Computing Series. Springer, Berlin, Germany, 2005.

[14] P. Rockett. The accuracy of sub-pixel localisation in the canny edge detector. In *British Machine Vision Conference (BMVC)*, 1999.

[15] S. Rusinkiewicz and M. Levoy. Efficient variants of the icp algorithm. In *International Conference on 3-D Digital Imaging and Modeling (3DIM)*, pages 145–152, Quebec City, Canada, 2001.

[16] G. C. Sharp, S. W. Lee, and D. K. Wehe. Invariant features and the registration of rigid bodies. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 932–937, 1999.

[17] L. Silva, O. R. P. Bellon, and K. L. Boyer. Robust multiview range image registration. In *Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI)*, pages 307–314, 2003.

[18] R. M. Steele and C. Jaynes. Feature uncertainty arising from covariant image noise. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 1063–1070, Washington, USA, 2005. IEEE Computer Society.

[19] R. Storn and K. Price. Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces. *Technical Report TR-95-012, ICSI*, 1995.

[20] O. Urfalıoğlu, P. Mikulastik, and I. Stegmann. Scale invariant robust registration of 3d-point data and a triangle mesh by global optimization. In *International Conference on Advanced Concepts for Intelligent Vision Systems (ACIVS)*, volume 127, pages 1059–1070, Antwerpen, Belgium, 2006.

[21] C.-C. Wang. Extrinsic calibration of a vision sensor mounted on a robot. *IEEE International Journal of Robotics and Automation (IJRA)*, 8(2):161–175, 1992.

[22] K. Zielinski, P. Weitkemper, R. Laur, and K.-D. Kammeyer. Examination of stopping criteria for differential evolution based on a power allocation problem. In *International Conference on Optimization of Electrical and Electronic Equipment (OPTIM)*, volume 3, pages 149–156, Brasov, Romania, 2006.

[23] K. Zielinski, P. Weitkemper, R. Laur, and K.-D. Kammeyer. Parameter study for differential evolution using a power allocation problem including interference cancellation. *IEEE Congress on Evolutionary Computation (CEC)*, pages 6748–6755, 2006.