# On Minimizing Errors in 3D Reconstruction for Stereo Camera Systems[1]

**S. Wenhardt[a], J. Denzler[b], and H. Niemann[a]**

[a] *Chair for Pattern Recognition, University of Erlangen–Nurnberg, Martensstr. 3, 91058 Erlangen, Germany*
[b] *Chair for Computer Vision Friedrich Schiller University of Jena, Ernst–Abbe–Platz 2, 07743 Jena, Germany*
*e-mail: {wenhardt, niemann}@informatik.uni–erlangen.de; denzler@informatik.uni–jena.de*

**Abstract**—Active reconstruction of 3D surfaces deals with the control of camera viewpoints to minimize error and uncertainty in the reconstructed shape of an object. In this paper we develop a mathematical relationship between the setup and focal lengths of a stereo camera system and the corresponding error in 3D reconstruction of a given surface. We explicitly model the noise in the image plane, which can be interpreted as pixel noise or as uncertainty in the localization of corresponding point features. The results can be used to plan sensor positioning, e.g., using information theoretic concepts for optimal sensor data selection.

**DOI:** 10.1134/S1054661807020228

## INTRODUCTION

In the past, more and more areas in computer vision gained advantages from active processing strategies, which means that the sensor data is acquired in an active, purposeful way. Examples are viewpoint selection for object recognition [1], actively control of the focal length during object tracking [2], and sequential sensor data selection for state estimation in general [4].

Besides these areas, up to now only a few approaches are known that suggest active sensor data selection for 3D reconstruction of surfaces and objects, for example, for range image data [5]. Obviously, for reconstructing the surface of an unknown object, the viewpoints of the recorded images strongly influence the resulting accuracy and robustness of the reconstruction. This observation is true, independent of the chosen approach for 3D reconstruction (stereo, factorization method, trifocal tensor). The quality mainly depends on the surface normal, the ex- and intrinsic parameters of the camera, and noise. So the question arises: is it possible to come up with a relationship between the selected views and the error and uncertainty of the reconstructed surface of an object? The long-term benefit of such an approach consists in the possibility of applying information theoretic methods for sequential sensor data selection [4] to 3D reconstruction as well. Towards that goal, in this paper we investigate the influence of the parameters of a stereo camera system on the error in reconstruction of a surface, explicitly taking into account the noise in the image acquisition and feature extraction process. To the best of our knowledge, such an investigation has not been done before.

The paper is structured as follows: first, we describe the setup for 3D reconstruction using triangulation in a normalized stereo camera system. Then we present a mathematical development of the reconstruction error in a simple 2D model, taking explicitly into account noise in a one dimensional image plane. We map the problem of optimal stereo positioning to an optimization problem. This will be analyzed first, to get the optimal focal length and the optimal baseline (translation in $x$ direction) in a normalized 2D stereo system. Further we gradually generalize this model, firstly by rotations, and secondly by translation in $x$ and $z$ directions. In this simple case we can perform a partial analytical analysis, but there are visibility assumptions which cannot be fulfilled in real stereo systems. Therefore, we further generalize to a 3D model with visibility constraints. In this model, we cannot perform an analytical analysis; therefore, we optimize the modifiable parameters by a Monte Carlo simulation. The results will be compared with the analytical results of the simple case. We present experimental results and compare them with the theoretical predictions, and conclude this paper with prospects for future study.

## PROBLEM OF 3D RECONSTRUCTION USING A NORMALIZED STEREO SYSTEM

First, we explain what we understand by a normalized stereo system: it consists of two cameras that share a common orientation. Translation is possible only in the $x$ direction (cf. Fig. 1). The points $O_\mathrm{l}$ and $O_\mathrm{r}$ are the optical centers. Each camera has its own coordinate system, with the $x$ and $z$ axes indexed by “l” for left camera and “r” for right camera. $\boldsymbol{t}_{\mathrm{lx}}$ and $\boldsymbol{t}_{\mathrm{rx}}$ are the translations of the cameras from the world coordinate system. The norm

$$\|\boldsymbol{t}\| := \|\boldsymbol{t}_{\mathrm{lx}} - \boldsymbol{t}_{\mathrm{rx}}\| = |t_{\mathrm{lx}} - t_{\mathrm{rx}}| \tag{1}$$

[1] The text was submitted by the authors in English.

Received April 5, 2005

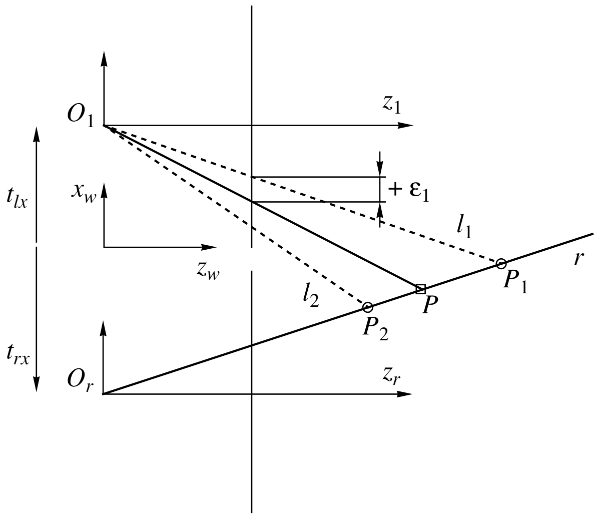

**Fig. 1.** Norm. Stereo system with errors by triangulation.

is called the baseline. For the triangulation, we have to know all camera parameters, i.e., the adjustable parameters (translation, focal length, and in the model without normalization the rotations [11]), the constant parameters (principal point, lens distortion, size of the pixels in millimeters in horizontal and vertical directions, the angle between the image axes [11, 12]), and the image coordinates. For real world data, disturbances occur, which results in a triangulation error. We analyze if there is a configuration of adjustable parameters for which the error is minimal.

## MODELING OF THE ERROR

There are many ways to model the disturbances and measure the error. Here we will assume that there is an error in one image plane (Fig. 1): the image coordinate of the projection of point $\boldsymbol{P}$ is disturbed by the value $\varepsilon_1$, e.g., caused by a nonaccurate solution of the correspondence problem. Hence, we select points in one image. These points are treated as being exact. Then we search for the corresponding points in the second image with a point-tracking algorithm [3, 10]. Thus, errors are assumed to occur only in the second image. We do not specify a statistical distribution of the error, but we model the maximal error, i.e., the worst case. Minimization of the error function means minimization of triangulation error if the maximal error occurs. Further on, the other parameters are assumed to be error free. For simplicity, all $y$ coordinates are set to zero, because in the plane, the lines cannot be skewed. We define the maximal error in the $x$ direction to be $\pm\varepsilon_1$ (cf. Fig. 1). We define the error $e$ as

$$e = \|\boldsymbol{P}_1 - \boldsymbol{P}_2\|^2, \tag{2}$$

where $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ are the triangulated points for the error $+\varepsilon_1$ and $-\varepsilon_1$, respectively.

An optimal 3D reconstruction means that we have to minimize the error function $e$ with respect to the free, adjustable parameters of our stereo camera system. For this, we have to derive the error function, i.e., to calculate the coordinates of point $\boldsymbol{P}_1$, which is the intersection of the lines of sight $r$ from the right camera system and the disturbed $l_1$ from the left (cf. Fig. 1). The linear equation for $r$ in the world coordinate system is

$$r : x_{\mathrm{w}} = -\frac{t_{\mathrm{rx}} - x_P}{z_P} z_{\mathrm{w}} + t_{\mathrm{rx}}, \tag{3}$$

where $(x_P,\ 0,\ z_P)$ are the coordinates of $\boldsymbol{P}$ the world coordinate system. With respect to the equations on perspective projection with focal length $f_1$, we can see that the linear equation for $l_1$ is

$$l_1 : x_{\mathrm{w}} = \left(-\frac{t_{\mathrm{lx}} - x_P}{z_P} + \frac{\varepsilon_1}{f_1}\right) z_{\mathrm{w}} + t_{\mathrm{lx}}. \tag{4}$$

From equations (3) and (4) we calculate $\boldsymbol{P}_1$:

$$\boldsymbol{P}_1 = \begin{pmatrix} \dfrac{(t_{\mathrm{lx}} - t_{\mathrm{rx}}) z_P f_1}{(t_{\mathrm{lx}} - t_{\mathrm{rx}}) f_1 - \varepsilon_1 z_P} \\ -\dfrac{(t_{\mathrm{lx}} - t_{\mathrm{rx}})(t_{\mathrm{rx}} - x_P) f_1}{(t_{\mathrm{lx}} - t_{\mathrm{rx}}) f_1 - \varepsilon_1 z_P} + t_{\mathrm{rx}} \end{pmatrix}. \tag{5}$$

The coordinates for point $\boldsymbol{P}_2$ can be calculated in the same way. Thus, for $e$ we get

$$e = \frac{4 f_1^2 \varepsilon_1^2 z_P^2 (t_{\mathrm{rx}} - t_{\mathrm{lx}})^2 ((t_{\mathrm{rx}} - x_P)^2 + z_P^2)}{((t_{\mathrm{rx}} - t_{\mathrm{lx}})^2 f_1^2 - \varepsilon_1^2 z_P^2)^2}. \tag{6}$$

## OPTIMIZATION OF FOCAL LENGTH

In our normalized stereo system, we can control focal length as well as translations in $x$ direction to improve the 3D reconstruction, i.e., to minimize the error function. If we modify any other parameter, the stereo system is no longer normalized. If we ignore the visibility, i.e., assuming infinite image planes, we can analyze all parameters separately. First, we analyze the influence of the focal length. Therefore, we differentiate $e$ with respect to the focal length $f_l$:

$$\frac{\partial e}{\partial f_l} \tag{7}$$
$$= \frac{z_P^2 \varepsilon_1^2 (t_{\mathrm{rx}} - t_{\mathrm{lx}})^2 ((t_{\mathrm{rx}} - x_P)^2 + z_P^2)((t_{\mathrm{rx}} - t_{\mathrm{lx}})^2 f_1^2 + z_P^2 \varepsilon_1^2)}{-0.125 f_1^{-1} ((t_{\mathrm{rx}} - t_{\mathrm{lx}})^2 f_1^2 - z_P^2 \varepsilon_1^2)^3}.$$

We can show that for $f_l \in\ ]\,0,\ z_P\varepsilon_1/(\mathrm{t}_{\mathrm{lx}} - t_{\mathrm{rx}})[$, the point $\boldsymbol{P}_1$ lies behind the cameras. So the relevant interval for the focal length is $f_l \in\ ]z_P\varepsilon_1/(t_{\mathrm{lx}} - t_{\mathrm{rx}}),\ \infty[$. For $f_l > z_P\varepsilon_1/(t_{\mathrm{lx}} - t_{\mathrm{rx}})$, the first derivative is negative; i.e., the error function $e$ is strictly monotonically decreasing and there is no minimum. We conclude that for a real camera system, the focal length should be chosen as large as possible, so that the object is just in the image, to improve the 3D reconstruction. This is also true for more than one point because the error function is then the sum of all errors (6) and the sum of monotonically decreasing functions is monotonically decreasing.

## OPTIMIZATION OF BASELINE (TRANSLATIONS IN *x* DIRECTION)

To minimize the error $e$, the gradient of $e$ has to be zero with respect to the translations $t_{lx}$ and $t_{rx}$, which are given in a fixed world coordinate system. We get a nonlinear system of equations with fifth-degree polynomials. This system of equations is generally not solvable by radicals [7], so we try to find a minimum by numerical optimization.

We search for a minimum with gradient descent method. In Fig. 2 we plotted ($t_{lx}$ $t_{rx}$), shown by different symbols for different initializations, and iterated 1000 times.

We observe that the translation $t_{rx}$ converges to a value near to zero and $t_{lx}$ becomes larger in each step. The trajectories converge to an asymptote. It seems to be the same asymptote for all tested initializations for different values of $z_P$, $f_l$ or $\varepsilon_l$; only for $x_P \neq 0$ it is shifted by $x_P$.

An already well-known result is that a larger baseline is better than a smaller one. In general, for $t_{lx} \longrightarrow \infty$, $e$ becomes zero:

$$\lim_{t_{lx} \to \infty} e = 0. \tag{8}$$

However, not only the length of the baseline is relevant for reconstruction: e.g., for $t_{lx} = -t_{rx} = 100$, $e = 28.8$ and for $t_{lx} = 110$, $t_{rx} = -10$, $e = 2.6$, although in the first case the baseline is twice as large. Further on, an infinite baseline does not imply that $e$ is zero:

$$\lim_{t_{rx} \to \infty} e = 4\varepsilon_l^2 z_P^2 / f_l^2. \tag{9}$$

So we conclude that in addition to the baseline, the position between cameras and points is also an important factor for 3D reconstruction.

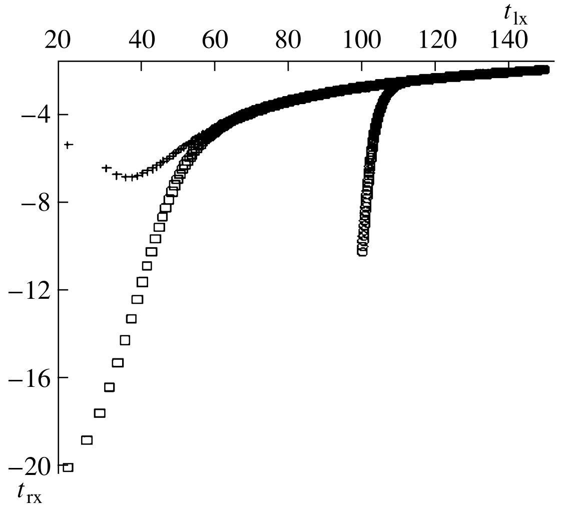

**Fig. 2.** Trajectories for translations: The initializations for ($t_{lx}$, $t_{rx}$) for the cross symbol is (20, –20), for the box it is (20, –5) and for circle it is (100, –10), under the assumptions $f_l = 1$, $P = (0\ 15)$, $\varepsilon_l = 1/2$.

If we want to reconstruct more than one point, the error is the sum of $e$ for the coordinates of different $\boldsymbol{P}_i$. The problem is more complex, because each error for one point depends on its coordinates ($x_{Pi}$, 0, $z_{Pi}$). We can see in Eq. (6) that $z_{Pi}$ has a strong influence on the value of $e$. Thus, points with large $z$ components result in a large error and they therefore have more influence on the minimization procedure.

## OPTIMIZATION OF ROTATION

We now want to generalize the model of the normalized stereo camera system, successively considering rotations which can be realized in practice using pan-and-tilt cameras. In our 2D model there is one rotation parameter for each camera. Therefore, we introduce a rotation about the $y$ axis which is perpendicular to the $x$–$z$ plane in Fig. 1. If the error is only in one camera, the rotation of the other is irrelevant. So we consider only rotation of the left camera by an angle $\alpha$. Then, the error function is

$$e = \frac{4\varepsilon_l^2 a((t_{lx} - x_P)\sin\alpha - z_P\cos\alpha)^4((t_{rx} - x_P)^2 + z_P^2)}{(a^2 - (b + c)^2)^2}, \tag{10}$$

where

$$\begin{aligned} a &= (t_{rx} - t_{lx})^2 f_l^2 z_P^2, \\ b &= \varepsilon_l((z_P\cos\alpha + x_P\sin\alpha)^2, \\ c &= -\varepsilon_l(t_{lx} + t_{rx})(x_P\sin^2\alpha + z_P\cos\alpha\sin\alpha) \\ &\quad + t_{lx}t_{rx}\sin^2\alpha). \end{aligned} \tag{11}$$

Differentiation of Eq. (10) with respect to $\alpha$ and computation of the zero crossings is possible. Due to lack of space, we have to omit the complicated derivation. We investigate the solution for $f_l = 1$, $\boldsymbol{P} = (0\ 0\ 15)$, $\varepsilon_l = 1/2$, $t_{lx} = 5$, $t_{rx} = -5$. For $\alpha = 0$ this is equivalent to the configuration of Fig. 3. There are two minima in $\alpha_1 = -1.89$ and $\alpha_2 = 1.25$ (values in radian). For $\alpha_1$, $\boldsymbol{P}_1$ is behind the camera. So the left camera must be rotated

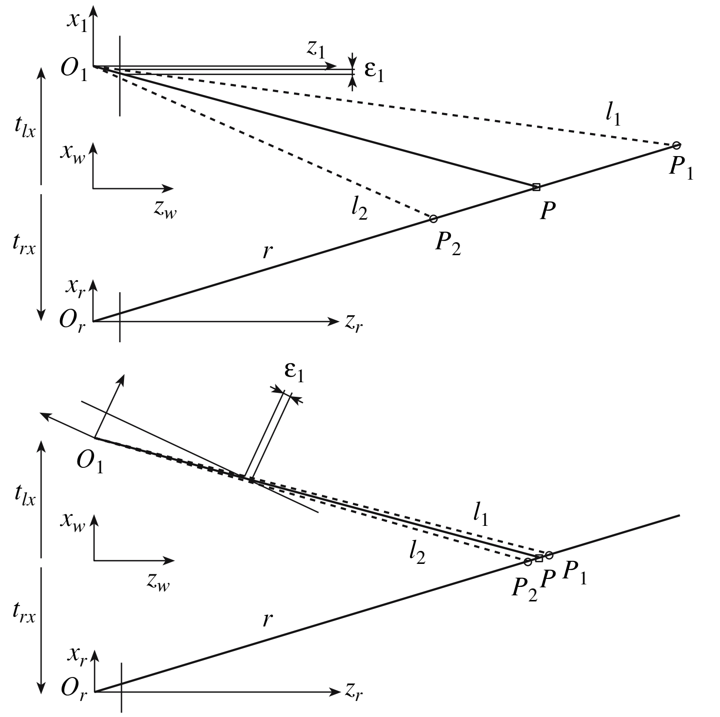

**Fig. 3.** The error of the start configuration (above) was decreased significantly by rotation of the left camera system (below), although the error $\varepsilon_1$ on the image plane is still the same.

counter clockwise by about 71°. Thus the camera should not be rotated toward, but turned away, while the object is in the image. The configuration with the rotated left camera by about 71° is shown in Fig. 3. One can see that the resulting triangulation error is less than in the start configuration (cf. Fig. 3), but we can also see that the configuration needs a large image plane if we do not want to lose the image point from the field of view. We will consider this problem in the generalized model introduced below.

Minimization by camera rotation for more than one point is similar to the case of translation: large values of $z_P$ result in a large $e$. Therefore, points at a larger distance have more influence on the minimization procedure and will bias the optimal solution for the rotation angle.

## OPTIMIZATION OF THE TRANSLATION IN THE *x* AND *z* DIRECTIONS

Before we start our analysis for the completely generalized model, we want to look again at the translation of the normalized stereo system, but now we also consider translations in $z$ direction. We have shown in the above section that a large baseline decreases the resulting triangulation error. Now we want to investigate whether the error also decreases if we modify the distance between the camera and the 3D point of the scene. The error function $e$ in this case is

$$e = \frac{4f_1^2\varepsilon_1^2(z_P - t_{\mathrm{lz}})^4((x_P - t_{\mathrm{rx}})^2 - (z_P - t_{\mathrm{lz}})^2)d^2}{(f_1^2d^2 - \varepsilon_1^2(z_P(z_P - t_{\mathrm{lz}} - t_{\mathrm{rz}}) + t_{\mathrm{lz}}t_{\mathrm{rz}})^2)^2}, \quad (12)$$

where

$$d = t_{\mathrm{lz}}(x_P - t_{\mathrm{rx}}) - t_{\mathrm{rz}}(x_P - t_{\mathrm{lx}}) + z_P(t_{\mathrm{rx}} - t_{\mathrm{lx}}), \quad (13)$$

and the four translation parameters $t_{\mathrm{lx}}$, $t_{\mathrm{rx}}$, $t_{\mathrm{lz}}$, and $t_{\mathrm{rz}}$ are as shown in Fig. 4.

Notice that if the translation in the $z$ direction is zero, we get the same error function $e$ as in Eq. (6).

Furthermore, $e$ depends of course on the focal length (of the left camera), the error $\varepsilon_l$, and on the coordinates of the 3D point $\boldsymbol{P} = (x_P\ 0\ z_P)$. Following the same argumentation as in the above section, we cannot search for the minimum of the error function $e$ analytically by searching for the zero of the derivative of $e$.

Therefore, we analyze the influence of the translation in the $x$ and $z$ directions by again using the gradient descent method.

We start with different initializations of the four translation parameters $t_{\mathrm{lx}}$, $t_{\mathrm{rx}}$, $t_{\mathrm{lz}}$, and $t_{\mathrm{rz}}$, and iterate 1000 times. We plotted ($t_{\mathrm{lx}}$, $t_{\mathrm{lz}}$) and ($t_{\mathrm{rx}}$, $t_{\mathrm{rz}}$) in Fig. 5, shown by different symbols for different initializations. There is a major difference to the above-mentioned case, in which there was optimization only in the $x$ direction:

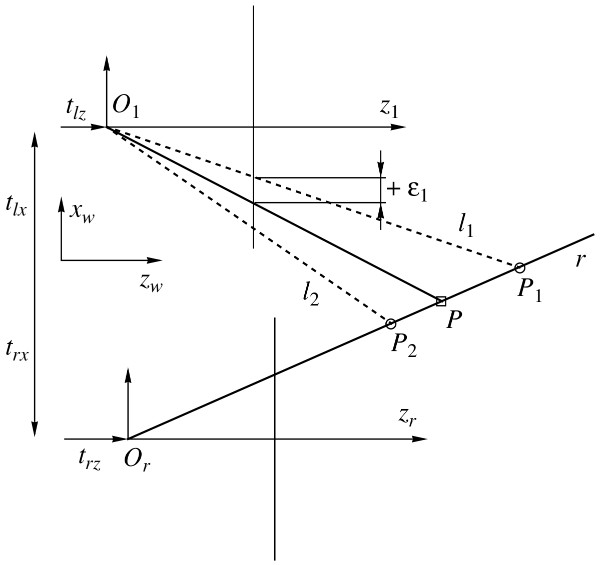

**Fig. 4.** Translation in $x$ and $z$ directions for the left and right camera system.

translation in the $x$ direction changes quite slowly, even for the left camera, but translation in the $z$ direction of the left camera is very large. Further on, the translations of the left camera are larger than of the right.

The left camera moves in the $z$ direction very close to the 3D point, while the baseline remains quite constant if we compare it with the case in which only the baseline can be modified. This effect happens even if the baseline is small in the initialization (cf. Fig. 5, box symbol).

The right camera moves in the $x$ direction again to the value of $x_P$ (which is zero in Fig. 5). This is the same behavior as in the former case. In the $z$ direction it moves slightly away from point $P$. The movement in the $x$ and $z$ directions can be explained, because it keeps the angle between the lines of sights of left and right camera from becoming too obtuse.

## INTERIM REMARKS

We now make some remarks on the previous results:

Our model was not a full 3D model, because we neglect the $y$ direction. Further, we assume that there is an error only in one camera (above it is always the left one). We modeled this error as an interval of uncertainty and analyzed the worst case scenario. The effects of each modifiable parameter were analyzed to give an idea of the influence of each parameter, which were focal length, baseline (translation only in the $x$ direction), rotation, and translation in the $x$ and $z$ directions. Of course, we cannot assume that optimizing the parameters separately we will obtain a global optimum for all parameters. Also, if we optimize one parameter after the other, we get a problem that up to now we have discussed only in brief: the problem of visibility, because the image plane is not infinite, or points lie behind the camera. For example, if we rotate the camera to turn away, we may need a very large image plane (cf. Fig. 3). If we increase the baseline in the configuration of Fig. 3, the point lies even behind the camera. Of course such configurations are not suitable for 3D reconstruction. The advantage of this analysis is that we can verify the results by plotting the configurations, and we can measure the length of the line $\mathbf{P}_1\,\mathbf{P}_2$. So we have

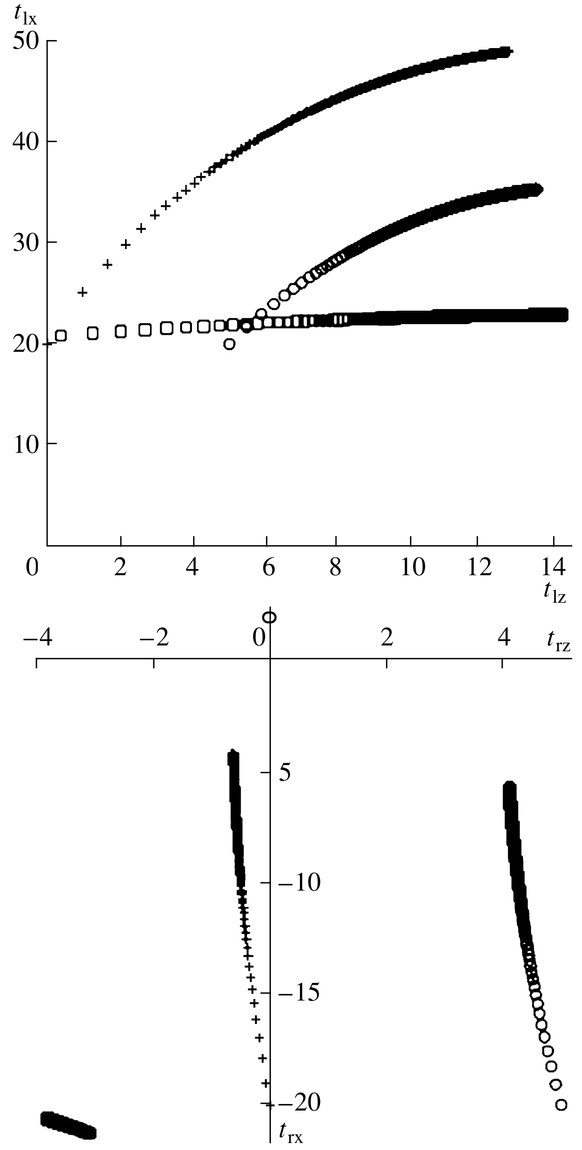

**Fig. 5.** Upper: trajectories of the left camera for different initializations ($t_{lx}$, $t_{lz}$): cross (20, 0); box (–2, 0), circle (20, 5).

Lower: trajectories of the right camera for the initializations:

($t_{rx}$, $t_{rz}$): cross (–20, 0); box (2, 0), circle (–20, 5)

The other under the assumptions are $f_l = 1$, $P = (0\ 15)$, $\varepsilon_1 = 1/2$.

verified the surprising result that turning the camera away from the object can reduce the triangulation error. Additionally, it allows us to derive analytical results for rotation and focal length.

## GENERALIZED MODEL

In the following part of this paper we will generalize our model: Firstly we assume that we have a projection from 3D space to the 2D image planes of the left and right camera. Secondly, on the image planes the coordinates are disturbed by additive normally distributed noise, i.e.,

$$\begin{aligned} x_{\mathrm{l}\varepsilon} &= x_{\mathrm{l}} + \varepsilon_1, \quad y_{\mathrm{l}\varepsilon} = y_{\mathrm{l}} + \varepsilon_2, \\ x_{\mathrm{r}\varepsilon} &= x_{\mathrm{r}} + \varepsilon_3, \quad y_{\mathrm{r}\varepsilon} = x_{\mathrm{r}} + \varepsilon_4 \end{aligned} \tag{14}$$

where $x_{\mathrm{l}}$ is the $x$ coordinate of the projected point $P$ on the image plane of the left camera; $x_{\mathrm{l}\varepsilon}y_{\mathrm{l}\varepsilon}$, $x_{\mathrm{r}\varepsilon}$ and $y_{\mathrm{r}\varepsilon}$ are the disturbed point coordinates; and

$$\varepsilon_1, \varepsilon_2, \varepsilon_3, \varepsilon_4 \sim N(0, \sigma^2) \tag{15}$$

are independent normally distributed values.

Since we are now considering the 3D case, we usually get skewed lines of sight, which we have to take care of during triangulation. As a consequence we get very complicated integrals if we calculate the expected value of the coordinates of the triangulated point. Thus, we decided to use Monte Carlo simulation for our further analysis.

Thirdly, we assume that the image planes are square with side length $s$. Finally, we now use $n$ points $P_i$, ($i$ = 1, …, $n$).

## MODELING OF THE ERROR

Similar to the simple model, we have to calculate an error function $e$, which measures the triangulation error of the $n$ points $P_i$, given the intrinsic and extrinsic parameters and specific values of $\varepsilon_1$, $\varepsilon_2$, $\varepsilon_3$, $\varepsilon_4$:

$$e(f_{\mathrm{l}}, f_{\mathrm{r}}, \mathbf{t}_{\mathrm{l}}, \mathbf{t}_{\mathrm{r}}, \mathbf{R}_{\mathrm{l}}, \mathbf{R}_{\mathrm{r}}, \varepsilon_1, \varepsilon_2, \varepsilon_3, \varepsilon_4) = \frac{1}{n}\sum_{i=1}^{n}\|P_i - P_i^*\| \tag{16}$$

where $P_i$ is the real 3D point, $P_i^*$ is the reconstructed point, $f$ is the focal length, $\mathbf{t}$ is the translation vector, $\mathbf{R}$ is the rotation matrix, subscript “l” denotes the left camera, and subscript “r” denotes the right camera.

We could use Eq. (16) if we know the specific values for the noise parameters $\varepsilon_1$, $\varepsilon_2$, $\varepsilon_3$, $\varepsilon_4$. However, we want to model the error if the noise parameters $\varepsilon_1$, $\varepsilon_2$, $\varepsilon_3$, $\varepsilon_4$ are normally distributed. Therefore, we calculate the expected error $\bar{e}$ of Eq. (16) by a Monte Carlo simulation with $m$ samples and get

$$\begin{aligned} \bar{e} &= \frac{1}{m}\sum_{j=1}^{m} e(f_{\mathrm{l}}, f_{\mathrm{r}}, \mathbf{t}_{\mathrm{l}}, \mathbf{t}_{\mathrm{r}}, \mathbf{R}_{\mathrm{l}}, \mathbf{R}_{\mathrm{r}}, \varepsilon_{1j}, \varepsilon_{2j}, \varepsilon_{3j}, \varepsilon_{4j}) \\ &= \frac{1}{mn}\sum_{j=1}^{m}\sum_{i=1}^{n}\|P_i - P_i^{(j)}\|, \end{aligned} \tag{17}$$

where $\varepsilon_{1j}$, $\varepsilon_{2j}$, $\varepsilon_{3j}$, $\varepsilon_{4j}$ are $m$ samples drawn from a normal distribution with zero mean and variance $\sigma^2$. We further assume that the variance $\sigma^2$ is independent of the intrinsic and extrinsic parameters.

To find the global minimum of Eq. (17), we have to solve a minimization problem with 14 parameters (the focal length, three translation and three rotation parameters for each camera) and some constraints (the 3D points have to lie in front of the camera, and their projections must be in the image plane, which we defined above as a finite square). We solve this optimization problem with boundary conditions by a sequential quadratic programming (SQP) method [6, 8, 9] as implemented in the Matlab optimization toolbox.

In order to see how this generalized model relates to the simple model, we first want to analyze the same cases of modifiable parameters, i.e., focal length, baseline (translation in the $x$ and $y$ directions), rotation, and translation in $x$, $y$, and $z$ direction. And last but not least, we want to analyze the optimization process, if all parameters are modified simultaneously.

The premises for the following Monte Carlo simulations are as follows: 25 points, which lie in a plane (as on a calibration pattern), should be reconstructed. We use this setting because we will verify our real experiments largely on a calibration pattern, which has two advantages: firstly, there are no self-occlusions of the object and, secondly, we get ground truth data in real experiments. The distance of each point to its neighbors on our virtual calibration pattern is 20 mm. The plane of the calibration pattern is parallel to the $x$–$y$ plane of the world coordinate system at a distance of 500 mm and the $z$ coordinate goes through the center of the pattern. The initializations of the camera parameters are

$$\begin{aligned} \mathbf{t}_{\mathrm{l}} &= (300, 0, 0)^{\mathrm{T}}, \\ \mathbf{t}_{\mathrm{r}} &= (-300, 0, 0)^{\mathrm{T}}, \\ \mathbf{R}_{\mathrm{l}} &= \mathbf{R}_{\mathrm{r}} = \mathbf{I}, \\ f_{\mathrm{l}} &= f_{\mathrm{r}} = 10, \end{aligned} \tag{18}$$

where $\mathbf{I}$ is the identity matrix. This configuration is illustrated in Fig. 6.

For the field of view constraint we assume a square image plane with a side length of 30 mm. Furthermore, we assume that the errors $\varepsilon_1$, $\varepsilon_2$, $\varepsilon_3$, $\varepsilon_4$ are independent and normally distributed with zero mean and variance.

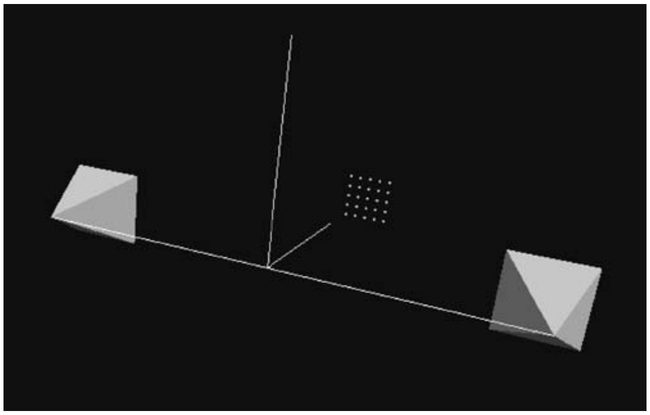

**Fig. 6.** Initial configuration for the Monte Carlo simulations. The pyramids represent the cameras: the tip is the optical center; the base is the image plane. The white lines are the axes of the world coordinate system.

$\sigma_2 = 0.2$ mm$^2$. We repeat the Monte Carlo simulation 20 times and give mean and the variance of the expected error $\bar{e}$. The mean of the expected error in this start configuration is $\bar{e} = 14.1$, the variance var($\bar{e}$) = $1.7 \times 10^{-3}$. Additionally, the computation time is given for one Monte Carlo simulation of each case. The simulation was done on an Intel Pentium 4 computer with 3.0 GHz and 1 GB RAM. The computation time is proportional to the number of 3D Points $P_i$.

## OPTIMIZATION OF FOCAL LENGTH

We now optimize only the focal length by using an SQP method to solve the minimization problem with constraints. The algorithm increases the focal lengths of both cameras, and in our configuration we get the optimal values

$$f_l = f_r = 44.1. \qquad (19)$$

These are the largest possible focal lengths, so that the calibration pattern is still completely in the image. The expected triangulation error $\bar{e}$ decreases from $\bar{e} = 14.1$ in the initial configuration to $\bar{e} = 3.2$, with a variance of var($\bar{e}$) = $5.9 \times 10^{-5}$.

We conclude that the larger the focal length, the smaller the triangulation error $\bar{e}$ in the general case. The observation in the 2D case, i.e., to use a focal length which is as large as possible so that the object is still completely visible, can be confirmed with this 3D experiment. Of course, the focal length is also bounded by the physical limits of the cameras.

The computation time in this case is about 2 min.

## OPTIMIZATION OF BASELINE (TRANSLATION IN *x* AND *y* DIRECTIONS)

In the simple case, the result was that the camera with errors on its image plane should be moved along the *x* axis away from the object to increase the baseline, while the camera without error should be moved close to the *z* axis (in the special configuration).

Since we have now errors on both image planes, we assume that both cameras behave like the left one in the simple case and indeed we get the optimal solution for the translation vectors

$$\mathbf{t}_l = (1460, 12, 0)^T \quad \mathbf{t}_r = (-1460, -10, 0)^T. \qquad (20)$$

The error is $\bar{e} = 9.25$, and its variance is var($\bar{e}$) = $6.7 \times 10^{-4}$ at the end of the optimization process. The

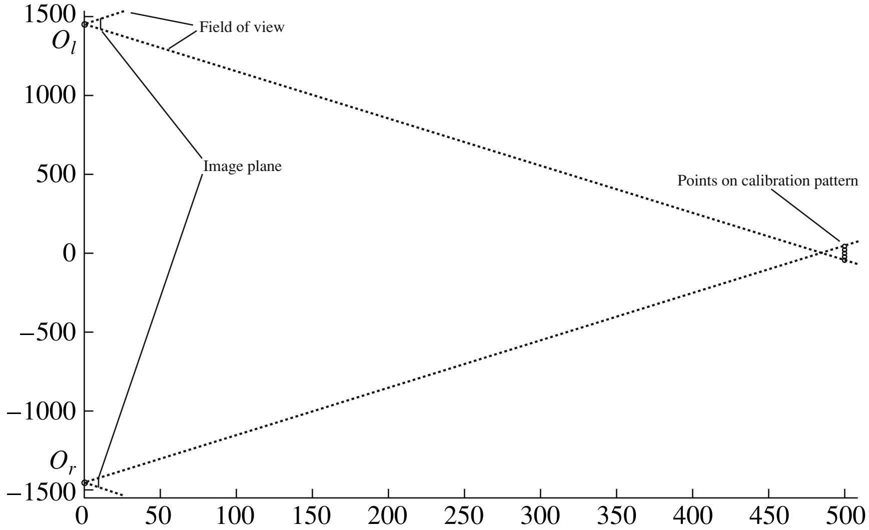

**Fig. 7.** End configuration after optimization of baseline with the Monte Carlo simulation.

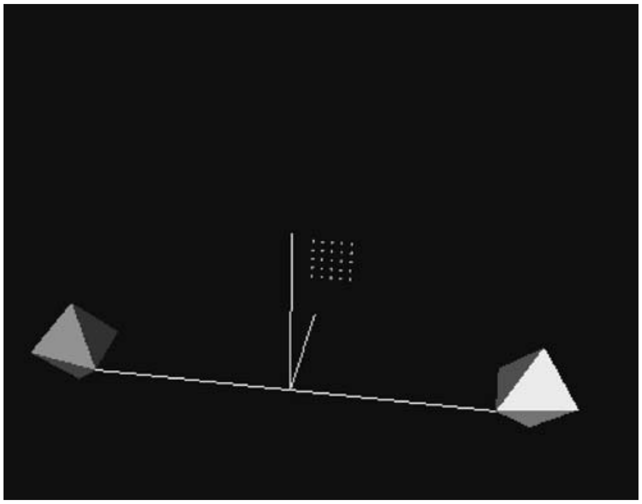

**Fig. 8.** End configuration, if we optimize the rotations in the generalized model. The cameras were turned away from and not towards the calibration pattern.

end configuration is shown in Fig. 7. The computation time in this case is about 8 min.

We notice that the baseline was increased (in the *x* direction), until at least one point of the calibration pattern lies at the edge of our image plane. We now discuss why the baseline was not further increased in the *y* direction either. To answer this question, we start with another initialization of the translation vectors:

$$\mathbf{t}_l = (300, 300, 0)^T \quad \mathbf{t}_r = (-300, -300, 0)^T. \quad (21)$$

In this case we get the optimal translation vectors:

$$\mathbf{t}_l = (1460, 1460, 0)^T \quad \mathbf{t}_r = (-1460, -1460, 0)^T. \quad (22)$$

The error decreases from the initialization with $\bar{e}$ = 12.0 to $\bar{e}$ = 9.1 in the end position. So we see that the error of the different initializations (18) and (21) is quite large, but at the end position (20) and (22), it is quite small. We assume that the error function is quite flat in this region, so the numeric algorithm does not optimize in the *y* direction.

We can verify this optimization result by starting with

$$\mathbf{t}_l = (0, 300, 0)^T \quad \mathbf{t}_r = (0, -300, 0)^T. \quad (23)$$

Indeed, we get the optimal solution at

$$\mathbf{t}_l = (11, 1460, 0)^T \quad \mathbf{t}_r = (-13, -1460, 0)^T, \quad (24)$$

and the error is $\bar{e}$ = 9.25. So this process is symmetric, too (if we ignore the slight asymmetry because of numerical inaccuracy).

So here we conclude that a larger baseline is better than a smaller one and the camera should be translated symmetrically to the object.

## OPTIMIZATION OF ROTATION

If we optimize the rotation angles, we still get a result similar to the simple case. Like the optimization of translation, again we have both cameras with noise at the image plane and therefore we get a symmetric result. The optimal rotation angles (cardan angles) are

$$\begin{aligned} \alpha_l &= \alpha_r = -0.77 \approx -44°, \\ \beta_l &= -\beta_r = 0.73 \approx 42°, \\ \gamma_l &= -\gamma_r = -0.01 \approx -0.8°, \end{aligned} \quad (25)$$

where $\alpha$ is the rotation about the *z* axis, $\beta$ about the *y* axis, $\gamma$ about the *x* axis, and the subscripts "l" and "r" denote the left and right cameras, respectively. Figure 8 shows the configuration with the points on the calibration pattern and the rotated cameras.

Here again the result shows that the camera should turn away and not towards the object as in the simple case. The rotation about the *z* axis has no influence on the expected error $\bar{e}$, because this causes only a rotation of the image in the image plane. The error was reduced to $\bar{e}$ = 2.7; the variance of $\bar{e}$ is var($\bar{e}$) = $1.6 \times 10^{-3}$. The computation time is about 18 min.

## OPTIMIZATION OF TRANSLATION (IN THE *x*, *y*, AND *z* DIRECTIONS)

We know from the simple case that the camera with noise in the image plane should be translated as close to the object as possible. Here in the generalized case we get similar results for the translation vectors:

$$\begin{aligned} \mathbf{t}_l &= (30.6, 3.3, 476.2)^T \\ \mathbf{t}_r &= (-28.7, -1.0, 477.1)^T. \end{aligned} \quad (26)$$

The slightly asymmetric solution is caused by numerical inaccuracy. The error of this configuration is $\bar{e}$ = 0.60, and its variance is var($\bar{e}$) = $7.9 \times 10^{-4}$. The computation time is 33 min. We can see again that the prediction of the simple model, i.e., reducing distance to the object, is true in the generalized case, too.

## OPTIMIZATION OF ALL PARAMETERS

Last but not least, we analyze the simultaneous optimization of all modifiable camera parameters. If we start with the initial configuration (see above) we get an error of $\bar{e}$ = 0.64. This must obviously be a local minimum and not a global one, because if we optimize only the translations, we get a value of $\bar{e}$ = 0.60.

Therefore, we start from the end positions of the above results ((19), (20), (25), (26)). If we start from the end position of the optimization of the translation

(Eq. (26)) we get an error of $\bar{e} = 0.48$ (variance: var($\bar{e}$) = $9.4 \times 10^{-2}$), and the optimal parameters are

$$\begin{gathered}\mathbf{t}_l = (51, 0, 461)^T \quad \mathbf{t}_r = (-42, -1, 477)^T, \\ \alpha_l = \alpha_r = 0 \approx 0°, \\ \beta_l = -\beta_r = -0.39 \approx -22°, \\ \gamma_l = -\gamma_r = -0.0 \approx -0.8°, \\ f_l = 30 \quad f_r = 38,\end{gathered} \tag{27}$$

However, we get the same error if we start the optimization of all parameters at the end position of the optimization of the baseline (20), although we get slightly different end positions:

$$\begin{gathered}\mathbf{t}_l = (50, 0, 450)^T \quad \mathbf{t}_r = (-50, 0, 450)^T, \\ \alpha_l = \alpha_r = 0 \approx 0°, \\ \beta_l = -\beta_r = 0.0 \approx 0°, \\ \gamma_l = -\gamma_r = 0.0 \approx 0°, \\ f_l = 37 \quad f_r = 37.\end{gathered} \tag{28}$$

Further, we can see that the function in this area is quite flat, because the solutions (27) and (28) are not so far away from each other and the error is approximately equal. Therefore, the numerical optimization process stops here.

We see here that the optimal position of the camera is very close to the object and in the above case the rotation is now toward the object. This is because without these rotations, the object is not completely in the image. The reduction of the error by turning the cameras away seems to have less influence than the translation of the cameras to the object.

Last but not least, we allude to the fact that modifying the focal length or the translation in the *z* direction does not have the same effect: an object becomes larger in the image if we increase the focal length or if we move the camera nearer to the object. However, in the first case, the angle between the lines of sight does not change, and in the second one, it becomes larger. Due to the fact that the angle is important for the triangulation result, we can explain why the focal length was only increased slightly in results (27) and (28).

The computation time in these cases is about 100 min.

The results of the presented cases in the generalized model are listed in Table 1. Of course, if we can modify all parameters, we get the smallest error, but the highest computation time. Further, increasing the baseline decreases the error, but in the other cases the error is much more reduced. Therefore, the modification of the baseline is not as important as rotation, focal length, or translation in the *z* direction. This is why the baseline even decreases if we only optimize translation in all directions. Only the decreasing baseline allows movement of the camera to the object without losing the object from the image.

**Table 1.** Results of the Monte Carlo simulation. Of course, if all parameters are modifiable, we get the best error, but the highest computation time

| | $\bar{e}$ | var($\bar{e}$) | Computation time |
|---|---|---|---|
| Initial configuration | 14.1 | $1.7 \times 10^{-3}$ | – |
| Focal length modifiable | 3.2 | $5.9 \times 10^{-5}$ | 2 min |
| Baseline modifiable | 9.25 | $6.7 \times 10^{-4}$ | 8 min |
| Rotation modifiable | 2.7 | $1.6 \times 10^{-3}$ | 18 min |
| Translation (*x*, *y*, *z* direction) modifiable | 0.60 | $7.9 \times 10^{-4}$ | 33 min |
| All parameters | 0.48 | $9.4 \times 10^{-2}$ | 100 min |

## EXPERIMENTAL RESULTS

In this section we present experimental results to show the influence of the adjustable camera parameters on the quality of the 3D reconstruction. We took images of a calibration pattern and a cube (cf. Fig. 9). We calibrated the cameras with the calibration pattern and reconstructed 49 points on it (experiment 1). In this case we can quantify the triangulation results based on ground truth data.

We did this in experiments with a calibration pattern for all theoretically analyzed combinations of modifiable parameters like above [focal length, baseline (translation in the *x* and *y* directions), rotation, translation (all directions), and all parameters].

In Tables 2 and 3 we present the results for modifying the focal length and the baseline. Further we reconstructed for these two cases all seven visible corners of the cube (cf. Fig. 9) and calculated the edge lengths, which we compared with the true value (experiment 2).

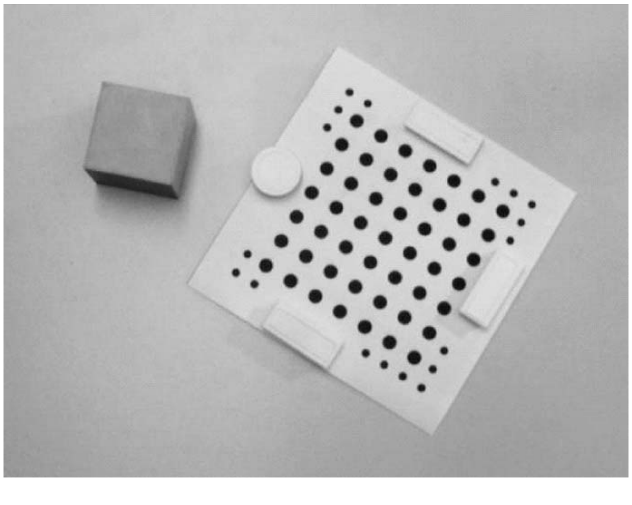

**Fig. 9.** Typical experiment image.

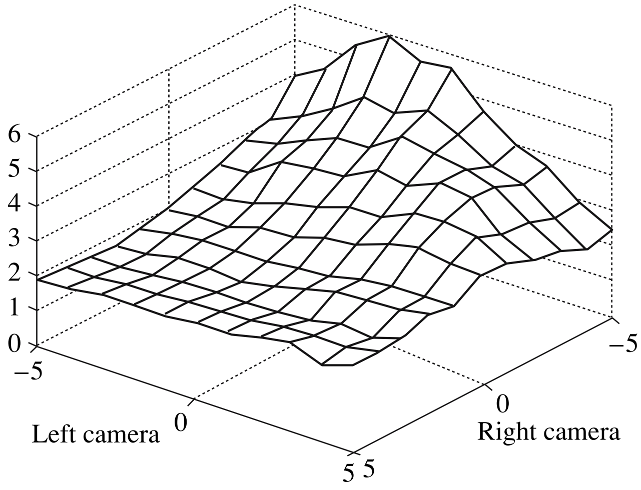

**Fig. 10.** The error function for different rotation angles for left and right camera. Again, left camera is turned away by –5° and the right one is turned away by +5°.

In Table 1, the first value in each cell is the mean difference between the real and reconstructed points in experiment 1. The second value is the mean difference of the measured edge lengths and the correct one (60 mm) in experiment 2. Table 3 lists more values for experiment 1 with the calibration pattern.

In the theory sections, we showed that if translation or focal length increases, the error decreases. So the largest errors are in the top left of Table 1, and the smallest ones are in the lower right; however, in experiment 2 there are 2 outliers (for $||t|| = 63, f_1 = 1487$ and $||t|| = 201, f_1 = 1155$). A possible reason for these outliers is that detection of the points not on the top side of the cube is quite inaccurate. However, if we ignore the outliers, we can see that the error decreases if the focal length increases (cf. columns of Table 2) or the baseline increases (cf. rows of Table 2). This verifies the theoretical results in practice.

**Table 2.** Experimental results for modifying baseline and focal length (focal length is in pixels, the other values in mm). The first value in each cell is the mean difference between the real and reconstructed points in experiment 1. The second value is the mean difference of the measured edge lengths and the correct one (60 mm) of the cube in experiment 2

| | $\|\|t\|\| = 51$ | $\|\|t\|\| = 63$ | $\|\|t\|\| = 201$ | $\|\|t\|\| = 326$ |
|---|---|---|---|---|
| $f_1 = 763$ | 6.8/28 | 4.5/25 | 1.5/9.9 | 1.0/5.6 |
| $f_1 = 1155$ | 1.1/13 | 1.0/8.3 | 0.4/2.2 | 0.3/2.3 |
| $f_1 = 1487$ | 0.8/0.8 | 0.6/0.11 | 0.3/0.73 | 0.2/0.4 |
| | $\|\|t\|\| = 35$ | $\|\|t\|\| = 67$ | $\|\|t\|\| = 105$ | |
| $f_1 = 771$ | 1.86 | 1.44 | 1.11 | |
| $f_1 = 1070$ | 1.72 | 0.98 | 0.95 | |
| $f_1 = 2400$ | 1.42 | – | – | |
| $f_1 = 2900$ | 0.78 | – | – | |

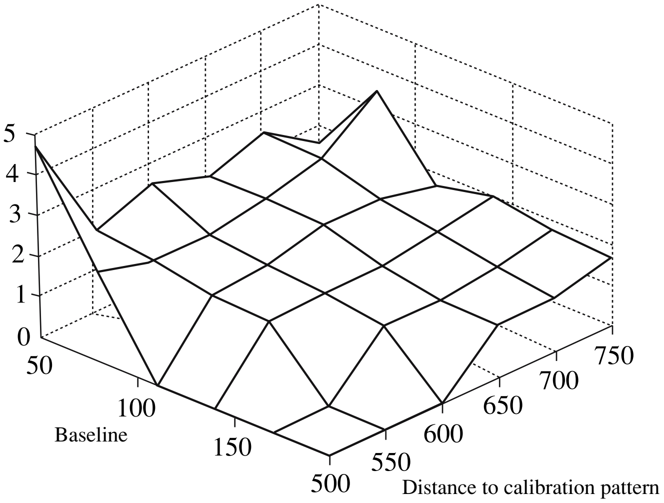

**Fig. 11.** The error depends on the baseline in a symmetric configuration and on the distance to the object. If the baseline was increased, the error decreases and if the distance to the object was decreased, the error is also decreasing. In the case, where the calibration pattern is partly out of the image we set the error function to zero, because correct camera calibration and thus triangulation is not possible.

Next we presented our results for the rotation parameters. As in the above considerations, a rotation about the $z$ axis has no influence on the position of the minimum, and the rotation about the $x$ axis should be small. Therefore, we took images only by rotating around the $y$ axis from –5° until 5° by steps of 1° for each camera. The results are presented in Fig. 10. We see that the rotation has a strong influence on the triangulation error, and turning the camera away decreases the error if we ignore the outliers. If we turn the camera towards the object, the error first increases, but, of course, the error decreases again if we turn in this direction more and more.

Now we want to present our results for the optimization of translation in $x$, $y$, and $z$ directions. As we saw in the theoretical investigations, translation in $z$ direction is important in addition to one in the $x$ or $y$ direc-

**Table 3.** A further experimental result for modifying baseline and focal length (focal length is in pixels, the other values in mm). The value in each cell is the mean difference between the real and reconstructed points in experiment 1 (reconstruction of a calibration pattern). In cells without a value, the calibration pattern was not completely visible

| | $\|\|t\|\| = 35$ | $\|\|t\|\| = 67$ | $\|\|t\|\| = 105$ |
|---|---|---|---|
| $f_1 = 771$ | 1.86 | 1.44 | 1.11 |
| $f_1 = 1070$ | 1.72 | 0.98 | 0.95 |
| $f_1 = 2400$ | 1.42 | – | – |
| $f_1 = 2900$ | 0.78 | – | – |

tions, and the respective third direction has less influence. Therefore, we analyze the influence of the $x$ and $z$ directions in symmetric cases. As we see in Fig. 11, the error is reduced if we decrease the distance to the object and/or we increase the baseline—neglecting outliers in the results.

## CONCLUSIONS

It is obvious that for 3D reconstruction not every recorded view is equally useful. We used a stereo system for our analysis and specified on which parameters the 3D reconstruction depends. There are unchangeable parameters and parameters controllable by an active vision system. The main question was what configuration of parameters results in a good triangulation.

First, in a simple 2D model we analyzed the influence of each parameter separately, where we can perform a partial analytical analysis. We were able to analytically prove that the error is strictly monotonically decreasing if the focal length increases. We also showed that a large baseline decreases the error, but the error also depends on the position between points and cameras. We further analyzed the effects of rotations. The result was that the camera should not turn to the object, but away from it. The last adjustable parameters in the simple case were the translation in the $x$ and $z$ directions. Here we could see that decreasing the distance between camera and object decreases the error and that the effect of increasing the baseline was smaller.

We generalized this simple model to a 3D model with visibility constraints and we assume errors for both cameras. In these cases we cannot perform analytical analysis. We performed a Monte Carlo evaluation instead, but the results of the simple case are conferrable to the generalized one if we optimize the parameters separately. Further, in the generalized case, we optimized all modifiable parameters simultaneously, but we may get only local minima of the error function. In general this cannot be avoided, but we can compare the results with the analytical cases of the simple model and try to find a good initialization for the nonlinear optimization process.

The presented model could also be used for different kinds of modifiable parameters for special uses. One example of use is a system of two zoom cameras mounted on a pan tilt unit. Such a system may be fixed in a room or mounted on a mobile platform. In each of these cases, we can define the modifiable parameters and start the optimization process.

In our future work we will try to determine the next best view if there are some already given images. Also we want to include the problem of visibility in the sense of self-occlusion of objects, which is a very important constraint in real applications. With these results we will be able to apply an already approved framework for optimal sensor data acquisition to the problem of active 3D reconstruction.

## REFERENCES

1. F. Deinzer, J. Denzler, and H. Niemann, “Viewpoint Selection—Planning Optimal Sequences of Views for Object Recognition,” in *Proc. of Conf. on Computer Analysis of Images and Patterns–CAIP’03*, Ed. by N. Petkov and M. A. Westenberg (Heidelberg, Springer, 2003), pp. 65–73.
2. J. Denzler, M. Zobel, and H. Niemann, “Information Theoretic Focal Length Selection for Real–Time Active 3D Object Tracking,” in *Proceedings of International Conference on Computer Vision*, pp. 400–407.
3. T. Zinsser, Ch. Graessl, and H. Niemann, “Effective Feature Tracking for Long Video Sequences,” in *Proc. of DAGM 2004*, pp. 326–333.
4. J. Denzler and C. M. Brown, “Information Theoretic Sensor Data Selection for Active Object Recognition and State Estimation”, IEEE Transactions on Pattern Analysis and Machine Intelligence, **24** (2), 145–157 (2002).
5. M. K. Reed and P. K. Allen, “Constrained–Based Sensor Planning for Scene Modeling,” IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (12), 1460–1466 (2000).
6. S. P. Han, “A Globally Convergent Method for Nonlinear Programming,” Journal of Optimization Theory and Applications **22**, 297 (1977).
7. N. Jacobsen, *Lectures in Abstract Algebra, Vol. III: Theory of Fields and Galois Theory* (D. van Nostrand Company, 1964).
8. M. J. D. Powell, “A Fast Algorithm for Nonlinearly Constrained Optimization Calculations,” in *Numerical Analysis*, Ed. by G. A. Watson, Lecture Notes in Mathematics, Vol. 630 (Springer, 1978).
9. M. J. D. Powell, “The Convergence of Variable Metric Methods For Nonlinearly Constrained Optimization Calculations,” in *Nonlinear Programming 3*, Eds. by O. L. Mangasarian, R. R. Meyer, and S. M. Robinson (Academic, 1978).
10. C. Tomasi and T. Kanade, “Detection and Tracking of Point Features,” in *Technical Report CMU–CS–91–132* (Carnegie Mellon University, 1991).

**Stefan Wenhardt,** born in 1978, graduated in mathematics at the University of Applied Sciences, Regensburg, Germany, in 2002, with a degree of Dipl.–Math.(FH). Since June 2002, he has been a research staff member at the Chair for Pattern Recognition at the Friedrich-Alexander-University of Erlangen–Nuremberg, Germany. The topics of his research are 3D reconstruction and active vision systems. He is author or coauthor of four publications.

**Joachim Denzler,** born April 16, 1967, received a degree of Diplom–Informatiker, Dr.–Ing. and Habilitation from the University of Erlangen in 1992, 1997, and 2003, respectively. Currently, he holds a position of a full professor for computer science and is head of the computer vision group, Faculty of Mathematics and Informatics, University of Jena. His research interests comprise active computer vision, object recognition and tracking, 3D reconstruction, and plenoptic modeling, as well as computer vision for autonomous systems. He is author and coauthor of over 80 journal papers and technical articles. He is member of the IEEE computer society, DAGM, and GI. For his work on object tracking, plenoptic modeling, and active object recognition and state estimation, he was awarded with the DAGM best paper awards in 1996, 1999, and 2001, respectively.

**Heinrich Niemann** obtained the degree of Dipl.–Ing. in Electrical Engineering and Dr.–Ing. from Technical University Hannover, Germany. He worked at the Fraunhofer Institut fur Informationsverarbeitung in Technik und Biologie, Karlsruhe, and at Fachhochschule Giessen in the department of Electrical Engineering. Since 1975 he has been Professor of Computer Science at the University of Erlangen–Nurnberg, where he was dean of the engineering faculty of the university from 1979–1981. From 1988–2000 he was head of the research group Knowledge Processing at the Bavarian Research Institute for Knowledge-based Systems (FORWISS). Since 1998 he has been the speaker of a special research area entitled Model-based Analysis and Visualization of Complex Scenes and Sensor Data, which is funded by the German Research Foundation (DFG). His fields of research are speech and image understanding and the application of artificial intelligence techniques in these fields. He is on the editorial boards of *Signal Processing*, *Pattern Recognition Letters*, *Pattern Recognition and Image Analysis*, and *Journal of Computing and Information Technology.* He is the author or coauthor of 7 books and about 400 journal and conference contributions, as well as editor or coeditor of 24 volumes of proceedings and special issues. He is a member of DAGM, ISCA, EURASIP, GI, and IEEE, and a Fellow of IAPR.