

Error Analysis of Stereo Calibration and Reconstruction

Agnieszka Bier¹ and Leszek Luchowski²

¹ Institute of Mathematics, Silesian University of Technology, Gliwice, Poland
`agnieszka.bier@polsl.pl`

² Institute of Theoretical and Applied Informatics, Polish Academy of Sciences,
Gliwice, Poland
`leszek.luchowski@iitis.gliwice.pl`

Abstract. This paper addresses the problem of the propagation of input data errors in the stereovision process and its influence on the quality of reconstructed 3D points. We consider only those particular camera calibration and 3D reconstruction algorithms which employ singular value decomposition (SVD) methods. Using the SVD Jacobian estimation method developed by Papadopoulos and Lourakis, we determine all the partial derivatives of outputs with respect to the inputs and present a set of tests applying them in various stereovision conditions in order to evaluate their impact on the quality of 3D reconstruction.

Keywords: Stereovision, sensitivity, projection matrix, 3D reconstruction.

1 Introduction

Many applications require stereovision algorithms to work within a given tolerance of error. The problem of precision in 3D reconstruction has been studied by many authors. For example, in [1] the authors investigate the error of a 3D point reconstructed by triangulation in the case of parallel image planes and derive the probability that the results are within a specified error margin. The precision analysis of 3D reconstruction from image sequences including the covariance matrix method and the evaluation of 3D reconstruction error have been thoroughly discussed in [6]. The latter is also considered in [4] in the sense of confidence intervals for the coordinates of the 3D point reconstructed from the cameras set in normal configuration and using disparity maps. Another approach is presented in [9], where the method of bounding boxes is used in the uncertainty analysis of 3D reconstruction. The sensitivity of 3D reconstruction of a specific kind of scene is analyzed in [2]. Hartley and Zisserman [3] analyzed the uncertainties in identifying a homography between two 2D images based on given points, using the best approximation in terms of the Mahalanobis distance between given and reconstructed coordinates. While not directly representing stereo vision, their

findings resemble ours (see Section 3) in that error increases with the distance of points from the origin of the coordinate system.

In this paper we present yet another approach to the problem by performing sensitivity analysis of SVD-based stereovision algorithms.

The paper is organized as follows. Section 2 introduces the calculation models for camera calibration and 3D reconstruction, and gives a detailed sensitivity analysis of both processes. In Section 3 we evaluate the sensitivity of the algorithm for different configurations of scene images varying with respect to camera positioning. The paper concludes with a discussion of the results.

2 Sensitivity Analysis of the Stereovision Process

2.1 Calculation Model

A 3D scene and its 2D image are mathematically represented as a set of pairs of corresponding 2D and 3D points, one being a *projection* of the other. Throughout this work, we will denote 2D coordinates with lowercase letters, and the corresponding 3D coordinates with capital letters. The homogeneous coordinates of $\mathbf{x} = [sx, sy, s]$ and $\mathbf{X} = [X, Y, Z, 1]$, where \mathbf{x} is the projection of \mathbf{X} , are related by the *projection matrix* \mathbf{P} :

$$s \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathbf{P} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \tag{1}$$

Calibration uses a set of known 3D points and their 2D images to determine the projection matrix. The equation (1) then leads to a system of equations which are linear with respect to entries of the projection matrix \mathbf{P} and hence can be written in the form:

$$\mathbf{M}\mathbf{R} = 0. \tag{2}$$

\mathbf{M} is a $2p \times 12$ matrix, where $p \geq 6$ is the number of pairs of points (x_i, y_i) and $(X_i, Y_i, Z_i, 1)$, $i = 1, 2, \dots, p$ used for calibration, and \mathbf{R} is a column vector composed of the entries of \mathbf{P} written row by row:

$$\mathbf{M} = \begin{bmatrix} \dots & \dots & \dots \\ \mathbf{X}_i & 0 & 0 & 0 & 0 & -x_i\mathbf{X}_i \\ 0 & 0 & 0 & 0 & \mathbf{X}_i & -y_i\mathbf{X}_i \\ \dots & \dots & \dots \end{bmatrix}, \tag{3}$$

$$\mathbf{R} = [\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3]^T. \tag{4}$$

The equation (2) usually does not have an exact nonzero solution, as in most cases it is over-determined. An approximate (least-squares) solution is found by using singular value decomposition.

The approximate solution \mathbf{R}_a to (2) is determined as

$$\mathbf{M} = \mathbf{U}\mathbf{D}\mathbf{V}^T \quad (\text{SVD}), \quad \mathbf{R}_a = [\mathbf{V}_{1,12}, \mathbf{V}_{2,12}, \dots, \mathbf{V}_{11,12}, \mathbf{V}_{12,12}]^T \quad (5)$$

\mathbf{R}_a minimizes the norm

$$\|\mathbf{MR}\| = \sum_{i=1}^{2p} (\mathbf{MR})_i^2$$

among all vectors \mathbf{R} of unit length.

To restore the matrix \mathbf{P} we need only to rearrange the entries of \mathbf{R}_a .

The reconstruction of the 3D scene is the process of recovering unknown 3D coordinates from $K \geq 2$ 2D images of the scene made from different points of view. We assume at least two calibrated cameras are used to take pictures of the scene, and the calibration stage performed for each of the cameras has yielded the projection matrices $\mathbf{P}^{(1)}, \mathbf{P}^{(2)}, \dots, \mathbf{P}^{(K)}$. Let $[x^{(i)}, y^{(i)}]$ be the Euclidean coordinates of the 2D projections of the unknown 3D point $[X, Y, Z]$, in the image produced by the camera with projection matrix $\mathbf{P}^{(i)}$ ($i = 1, 2, \dots, K$).

Recalling equation (1) for each camera and the pair of 2D and 3D coordinate vectors, we obtain a system of equations

$$s^{(i)} \begin{bmatrix} x^{(i)} \\ y^{(i)} \\ 1 \end{bmatrix} = \mathbf{P}^{(i)} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \quad i = 1, 2, \dots, K \quad (6)$$

Usually, having $K \geq 2$ views ensures that the system of equations (6) is (over)determined with respect to the unknown 3D coordinates, and therefore it has a unique (possibly least-squares approximate) solution. An under-determined system of equations can result from inappropriate camera positioning and it will not be covered here. We will, however, deal with "almost-under-determined" systems, which appear to be quite frequent in real life 3D reconstruction processes.

After rearranging the system of equations (6), we obtain the following matrix equation:

$$\mathbf{L} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \mathbf{B} \quad (7)$$

for \mathbf{L} being a matrix of dimension $2K \times 3$:

$$L = \begin{pmatrix} \mathbf{P}_{1,1}^{(1)} - x^{(1)}\mathbf{P}_{3,1}^{(1)} & \mathbf{P}_{1,2}^{(1)} - x^{(1)}\mathbf{P}_{3,2}^{(1)} & \mathbf{P}_{1,3}^{(1)} - x^{(1)}\mathbf{P}_{3,3}^{(1)} \\ \mathbf{P}_{2,1}^{(1)} - y^{(1)}\mathbf{P}_{3,1}^{(1)} & \mathbf{P}_{2,2}^{(1)} - y^{(1)}\mathbf{P}_{3,2}^{(1)} & \mathbf{P}_{2,3}^{(1)} - y^{(1)}\mathbf{P}_{3,3}^{(1)} \\ \mathbf{P}_{1,1}^{(2)} - x^{(2)}\mathbf{P}_{3,1}^{(2)} & \mathbf{P}_{1,2}^{(2)} - x^{(2)}\mathbf{P}_{3,2}^{(2)} & \mathbf{P}_{1,3}^{(2)} - x^{(2)}\mathbf{P}_{3,3}^{(2)} \\ \mathbf{P}_{2,1}^{(2)} - y^{(2)}\mathbf{P}_{3,1}^{(2)} & \mathbf{P}_{2,2}^{(2)} - y^{(2)}\mathbf{P}_{3,2}^{(2)} & \mathbf{P}_{2,3}^{(2)} - y^{(2)}\mathbf{P}_{3,3}^{(2)} \\ \dots & \dots & \dots \\ \mathbf{P}_{2,1}^{(K)} - y^{(K)}\mathbf{P}_{3,1}^{(K)} & \mathbf{P}_{2,2}^{(K)} - y^{(K)}\mathbf{P}_{3,2}^{(K)} & \mathbf{P}_{2,3}^{(K)} - y^{(K)}\mathbf{P}_{3,3}^{(K)} \end{pmatrix},$$

and \mathbf{B} the column vector of size $2K$:

$$\mathbf{B} = \begin{bmatrix} x^{(1)}\mathbf{P}_{3,4}^{(1)} - \mathbf{P}_{1,4}^{(1)} \\ y^{(1)}\mathbf{P}_{3,4}^{(1)} - \mathbf{P}_{2,4}^{(1)} \\ x^{(2)}\mathbf{P}_{3,4}^{(2)} - \mathbf{P}_{1,4}^{(2)} \\ y^{(2)}\mathbf{P}_{3,4}^{(2)} - \mathbf{P}_{2,4}^{(2)} \\ \vdots \\ y^{(K)}\mathbf{P}_{3,4}^{(K)} - \mathbf{P}_{2,4}^{(K)} \end{bmatrix}.$$

Equation (7) can be solved for the unknown Euclidean vector $[X, Y, Z]$ using the pseudo-inverse matrix, which leads to the least-squares solution:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_{ls} = \mathbf{L}^+ \mathbf{B}, \tag{8}$$

where \mathbf{L}^+ denotes the Moore-Penrose pseudo-inverse of matrix \mathbf{L} . The vector $[X, Y, Z]_{ls}^T$ minimizes the least-squares norm:

$$\sum_{j=1}^{2K} ((\mathbf{L}[X, Y, Z]^T)_j - \mathbf{B}_j)^2.$$

2.2 Sensitivity of Camera Calibration

At the calibration stage we assume that the 3D coordinates of the points \mathbf{X}_i are known accurately, while the 2D coordinates of their projection \mathbf{x}_i are read from the images with some error. We give here the analysis of propagation of the measurement errors to the entries of the resulting projection matrix.

In Section 2.1 we calibrated the cameras using the SVD method to solve a system of linear equations (2). Now we will examine the sensitivity of this method to input errors. The dependencies of SVD outputs (the entries of the three matrix components) on the entries of the input matrix were thoroughly discussed in [7]. The method proposed by Papadopoulo and Lourakis allows the Jacobian of the SVD components to be determined with respect to the entries of the matrix being decomposed, considering the SVD as a transformation of the matrix entries. We apply this method to estimate calibration error. Assume that $M = [m_{ij}]$ defined in Section 2.1 contains both accurate inputs and error-burdened ones. Let M have the SV decomposition defined in (5). Then, following Equation (9) in [7], we have:

$$\frac{\partial \mathbf{V}}{\partial m_{i,j}} = -\mathbf{V}\Omega_{\mathbf{V}}^{ij}, \quad \frac{\partial \mathbf{U}}{\partial m_{i,j}} = \mathbf{U}\Omega_{\mathbf{U}}^{ij},$$

where $\Omega_{\mathbf{U}}^{ij}$ and $\Omega_{\mathbf{V}}^{ij}$ are matrices of size $2p \times 2p$ and 12×12 , respectively, and their entries can be determined as solutions of the following systems:

$$\begin{cases} d_l \Omega_{\mathbf{U}}^{ij}{}_{kl} + d_k \Omega_{\mathbf{V}}^{ij}{}_{kl} = u_{ik} v_{jl}, \\ d_k \Omega_{\mathbf{U}}^{ij}{}_{kl} + d_l \Omega_{\mathbf{V}}^{ij}{}_{kl} = -u_{il} v_{jk}. \end{cases} \tag{9}$$

In this notation, d_i is the i -th diagonal entry of the D component in SVD, while $u_{i,j}$ and $v_{i,j}$ are the entries of matrices \mathbf{U} and \mathbf{V} .

If $d_k \neq d_l$, then the obtained entries of matrix $\Omega_{\mathbf{V}}^{ij}$ have the form:

$$\Omega_{\mathbf{V}}^{ij} = \frac{d_l u_{il} v_{jk} + d_k u_{ik} v_{jl}}{d_k^2 - d_l^2}$$

If two or more singular values are equal, then - as suggested in [7] - the equations (9) related to those values have to be solved with a least squares method in order to obtain the Jacobian with the smallest possible norm. Then the partial derivatives can be determined from the equation:

$$\frac{\partial v_{kl}}{\partial m_{ij}} = - \sum_{s=1}^{12} v_{ks} \Omega_{\mathbf{V}}^{ij} v_{sl}. \quad (10)$$

Returning to the notation of the projection matrix we have:

$$\frac{\partial \mathbf{P}_{\mathbf{k},1}}{\partial m_{i,j}} = \frac{\partial v_{4(k-1)+l,12}}{\partial m_{i,j}}, \quad (11)$$

hence, if (x_s, y_s) is the 2D image of the s -th 3D point $(X_s, Y_s, Z_s, 1)$,

$$\begin{aligned} \frac{\partial \mathbf{P}_{\mathbf{k},1}}{\partial x_s} &= \sum_{(i,j)} \frac{\partial \mathbf{P}_{\mathbf{k},1}}{\partial m_{i,j}} \cdot \frac{\partial m_{i,j}}{\partial x_s} = \\ &= -X_s \cdot \frac{\partial \mathbf{P}_{\mathbf{k},1}}{\partial m_{2s-1,9}} - Y_s \cdot \frac{\partial \mathbf{P}_{\mathbf{k},1}}{\partial m_{2s-1,10}} - Z_s \cdot \frac{\partial \mathbf{P}_{\mathbf{k},1}}{\partial m_{2s-1,11}} - \frac{\partial \mathbf{P}_{\mathbf{k},1}}{\partial m_{2s-1,12}} = \\ &= -X_s \cdot \frac{\partial v_{4(k-1)+l,12}}{\partial m_{2s-1,9}} - Y_s \cdot \frac{\partial v_{4(k-1)+l,12}}{\partial m_{2s-1,10}} - Z_s \cdot \frac{\partial v_{4(k-1)+l,12}}{\partial m_{2s-1,11}} - \\ &\quad - \frac{\partial v_{4(k-1)+l,12}}{\partial m_{2s-1,12}} \end{aligned} \quad (12)$$

and

$$\begin{aligned} \frac{\partial \mathbf{P}_{\mathbf{k},1}}{\partial y_s} &= \sum_{(i,j)} \frac{\partial \mathbf{P}_{\mathbf{k},1}}{\partial m_{i,j}} \cdot \frac{\partial m_{i,j}}{\partial y_s} = \\ &= -X_s \cdot \frac{\partial v_{4(k-1)+l,12}}{\partial m_{2s,9}} - Y_s \cdot \frac{\partial v_{4(k-1)+l,12}}{\partial m_{2s,10}} - Z_s \cdot \frac{\partial v_{4(k-1)+l,12}}{\partial m_{2s,11}} - \\ &\quad - \frac{\partial v_{4(k-1)+l,12}}{\partial m_{2s,12}}, \end{aligned} \quad (13)$$

where the partial derivatives $\frac{\partial v_{4(k-1)+l,12}}{\partial m_{2s-1,9}}$, $\frac{\partial v_{4(k-1)+l,12}}{\partial m_{2s-1,10}}$, $\frac{\partial v_{4(k-1)+l,12}}{\partial m_{2s-1,11}}$, $\frac{\partial v_{4(k-1)+l,12}}{\partial m_{2s-1,12}}$, $\frac{\partial v_{4(k-1)+l,12}}{\partial m_{2s,9}}$, $\frac{\partial v_{4(k-1)+l,12}}{\partial m_{2s,10}}$, $\frac{\partial v_{4(k-1)+l,12}}{\partial m_{2s,11}}$ and $\frac{\partial v_{4(k-1)+l,12}}{\partial m_{2s,12}}$ are determined from equations (10).

2.3 Sensitivity of 3D Reconstruction

Having determined the projection matrices of the cameras and given the 2D coordinates, in the images, of a feature point, we can use formula (8) to recover its 3D coordinates. Note that projection matrices as well as 2D coordinates are burdened with errors. The uncertainty of the entries of the projection matrices

were derived in the previous section. For now, we assume that the projection matrices are given with their uncertainties as $\mathbf{P}^{(i)} \pm \Delta \mathbf{P}^{(i)}$ and we are not interested in the source of these errors. At the same time, we consider the errors of reading the 2D coordinates $(x^{(i)}, y^{(i)})$ of a feature point $\bar{\mathbf{X}} = [X, Y, Z]^T$.

Recall the solution (8) to the 3D reconstruction problem given in section 2.1. We are now interested in the partial derivatives of the reconstructed coordinates with respect to any of the inputs that can be burdened with errors.

$$\frac{\partial \bar{\mathbf{X}}}{\partial r} = \frac{\partial (\mathbf{L} + \mathbf{B})}{\partial r} = \frac{\partial \mathbf{L}^+}{\partial r} \mathbf{B} + \mathbf{L}^+ \frac{\partial \mathbf{B}}{\partial r}, \tag{14}$$

where r represents any of the parameters $\mathbf{P}_{k,1}, x^{(i)}, y^{(i)}$. Clearly, vector $\frac{\partial \mathbf{B}}{\partial r}$ can be determined directly from the vector \mathbf{B} , as for $i = 1, 2, \dots, K$ we have

$$\begin{aligned} \frac{\partial \mathbf{B}_{2i-1}}{\partial \mathbf{P}_{3,4}^{(i)}} &= x^{(i)}, & \frac{\partial \mathbf{B}_{2i-1}}{\partial \mathbf{P}_{1,4}^{(i)}} &= -1, & \frac{\partial \mathbf{B}_{2i}}{\partial \mathbf{P}_{3,4}^{(i)}} &= y^{(i)}, & \frac{\partial \mathbf{B}_{2i}}{\partial \mathbf{P}_{1,4}^{(i)}} &= -1, \\ \frac{\partial \mathbf{B}_{2i-1}}{\partial x^{(i)}} &= \mathbf{P}_{3,4}^{(i)}, & \frac{\partial \mathbf{B}_{2i}}{\partial y^{(i)}} &= \mathbf{P}_{3,4}^{(i)}, \end{aligned}$$

and

$$\frac{\partial \mathbf{B}_j}{\partial \mathbf{P}_{k,1}^{(i)}} = \frac{\partial \mathbf{B}_j}{\partial x^{(i)}} = \frac{\partial \mathbf{B}_j}{\partial y^{(i)}} = 0$$

elsewhere. Likewise, the derivatives for the entries of the matrix \mathbf{L} are:

$$\begin{aligned} \frac{\partial \mathbf{L}_{2i-1,j}}{\partial \mathbf{P}_{1,j}^{(i)}} &= 1, & \frac{\partial \mathbf{L}_{2i,j}}{\partial \mathbf{P}_{2,j}^{(i)}} &= 1, & \frac{\partial \mathbf{L}_{2i-1,j}}{\partial \mathbf{P}_{3,j}^{(i)}} &= -x^{(i)}, & \frac{\partial \mathbf{L}_{2i,j}}{\partial \mathbf{P}_{3,j}^{(i)}} &= -y^{(i)}, & j &= 1, 2, 3, \\ \frac{\partial \mathbf{L}_{2i-1,1}}{\partial x^{(i)}} &= -\mathbf{P}_{3,1}^{(i)}, & \frac{\partial \mathbf{L}_{2i-1,2}}{\partial x^{(i)}} &= -\mathbf{P}_{3,2}^{(i)}, & \frac{\partial \mathbf{L}_{2i-1,3}}{\partial x^{(i)}} &= -\mathbf{P}_{3,3}^{(i)}, \\ \frac{\partial \mathbf{L}_{2i,1}}{\partial y^{(i)}} &= -\mathbf{P}_{3,1}^{(i)}, & \frac{\partial \mathbf{L}_{2i,2}}{\partial y^{(i)}} &= -\mathbf{P}_{3,2}^{(i)}, & \frac{\partial \mathbf{L}_{2i,3}}{\partial y^{(i)}} &= -\mathbf{P}_{3,3}^{(i)}, \end{aligned}$$

while all other derivatives $\frac{\partial \mathbf{L}_{s,t}}{\partial \mathbf{P}_{k,1}^{(i)}}, \frac{\partial \mathbf{L}_{s,t}}{\partial x^{(i)}}, \frac{\partial \mathbf{L}_{s,t}}{\partial y^{(i)}}$ are zeros.

The study of $\frac{\partial \mathbf{L}^+}{\partial r}$ needs more complicated analysis. Recall from section 2.1 that $\mathbf{L}^+ = \mathbf{V} \Sigma^+ \mathbf{U}^T$, hence for every pair of indices (i, j) :

$$\frac{\partial \mathbf{L}^+}{\partial l_{i,j}} = \frac{\partial \mathbf{V}}{\partial l_{i,j}} \Sigma^+ \mathbf{U}^T + \mathbf{V} \frac{\partial \Sigma^+}{\partial l_{i,j}} \mathbf{U}^T + \mathbf{V} \Sigma^+ \frac{\partial \mathbf{U}^T}{\partial l_{i,j}}. \tag{15}$$

Since all entries of the diagonal matrix Σ^+ are the reciprocals of the entries of matrix Σ , except for those equal to zero, which remain unchanged, we obtain:

$$\frac{\partial \Sigma^+_{k,k}}{\partial l_{i,j}} = -\frac{1}{(\Sigma_{k,k})^2} \cdot \frac{\partial \Sigma_{k,k}}{\partial l_{i,j}},$$

if $\Sigma_{k,k} \neq 0$, and

$$\frac{\partial \Sigma^+_{k,k}}{\partial l_{i,j}} = 0$$

otherwise. Additionally, for the case of $\Sigma_{k,k} \neq 0$, following the relations in [7] we have:

$$\frac{\partial \Sigma^+_{k,k}}{\partial l_{i,j}} = -\frac{1}{(\Sigma_{k,k})^2} \cdot \mathbf{U}_{i,k} \mathbf{V}_{j,k}.$$

The rest we get from [7]:

$$\frac{\partial \mathbf{U}}{\partial l_{i,j}} = \mathbf{U} \Omega_{\mathbf{U}}^{ij}, \quad \frac{\partial \mathbf{V}}{\partial l_{i,j}} = -\mathbf{V} \Omega_{\mathbf{V}}^{ij},$$

where $\Omega_{\mathbf{U}}^{ij}$ and $\Omega_{\mathbf{V}}^{ij}$ are matrices defined for the decomposition $\mathbf{L} = \mathbf{U} \Sigma \mathbf{V}^T$ and their entries can be obtained from the systems of equations analogous to (9).

Applying all these relations to the formula (15), we determine the derivatives of the entries of \mathbf{L}^+ with respect to particular entries of matrix \mathbf{L} . Now, for parameter r , which represents one of parameters $\mathbf{P}_{\mathbf{k},1}$, $x^{(i)}$, $y^{(i)}$, we obtain:

$$\frac{\partial \mathbf{L}^+}{\partial r} = \frac{\partial \mathbf{L}^+}{\partial \mathbf{L}} \cdot \frac{\partial \mathbf{L}}{\partial r}, \tag{16}$$

and more precisely

$$\frac{\partial \mathbf{L}^+}{\partial r} = \sum_{(i,j)} \frac{\partial \mathbf{L}^+}{\partial L_{i,j}} \cdot \frac{\partial L_{i,j}}{\partial r}.$$

Finally, we have all the derivatives needed to compute $\frac{\partial \bar{\mathbf{X}}}{\partial r}$ using equation (14):

$$\frac{\partial \bar{\mathbf{X}}}{\partial r} = \left(\sum_{(i,j)} \frac{\partial \mathbf{L}^+}{\partial L_{i,j}} \cdot \frac{\partial L_{i,j}}{\partial r} \right) \mathbf{B} + \mathbf{L}^+ \frac{\partial \mathbf{B}}{\partial r}. \tag{17}$$

3 Practical Applications

The purpose of the tests performed on the simulated 3D scene and its images is to confront the theory presented in the previous section with an actual stereovision process. We used POV-Ray (Persistence of Vision Ray-Tracer) to generate photo-realistic images from descriptions of scenes and camera positions. An important advantage of a virtual scene is that accurate 3D coordinates are known and can be used as a reference when evaluating the results of 3D reconstruction.

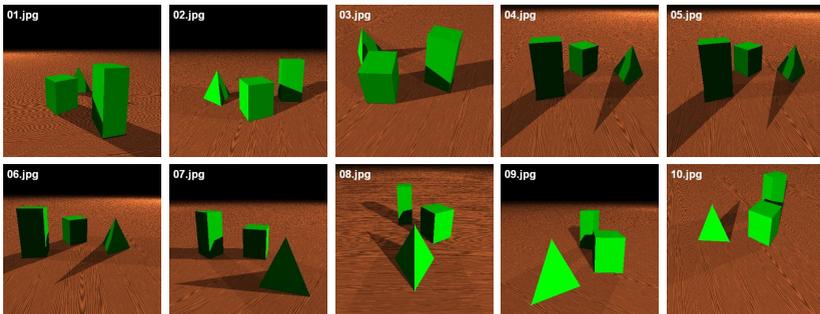


Fig. 1. Images obtained in POV-Ray

Our test scene consisted of a tetrahedron, a cuboid and a cube, all of a similar size, standing on a plane. For testing purposes, 11 pictures of the scene were created in POVray, each from a different point of view and camera direction. Ten of them are presented in Figure 1. The eleventh is shown in Figure 2 along with the labeling of the vertices.

All the resulting images were next subjected to calibration. The seven visible vertices of the cube served as calibration points. Using formulas 12 and 13 derived in the previous section, we determined the derivatives $\frac{\partial \mathbf{P}^{(i)}}{\partial x_j}$ and $\frac{\partial \mathbf{P}^{(i)}}{\partial y_j}$, which are the derivative matrices of the i -th projection matrix with respect to the x and y coordinate of the j -th calibration point in the i -th image.

In the following analysis, the measure of the sensitivity $\mathcal{S}_j(P^{(i)})$ of the projection matrix $\mathbf{P}^{(i)}$ to error in the j -th calibration point is defined as the sum of squares of all entries of matrices $\frac{\partial \mathbf{P}^{(i)}}{\partial x_j}$ and $\frac{\partial \mathbf{P}^{(i)}}{\partial y_j}$:

$$\mathcal{S}_j(P^{(i)}) := \sum_{k,l} \left[\left(\frac{\partial \mathbf{P}_{k,l}^{(i)}}{\partial x_j} \right)^2 + \left(\frac{\partial \mathbf{P}_{k,l}^{(i)}}{\partial y_j} \right)^2 \right].$$

Such definition seems reasonable as it gathers the influence of both coordinates on the whole projection matrix. It has, however, some drawbacks. As the components in the sum are taken with equal weights, their impact on further processing is not taken into account. This sum cannot therefore be regarded as a measure of calibration quality, which should be considered from the point of view of the quality of the whole stereovision process.

Table 1 presents the sensitivities of projection matrices to all calibration points separately and combined sensitivity to errors in coordinates of all calibration points, calculated as square root of the sum of squares of sensitivities to particular points.

The implementation of the differential method presented in the previous section allowed us to determine those of all eleven projection matrices which are the most sensitive to input errors, and those calibration points which influence the most and the least the precision of calibration. One can observe that some calibration points have significantly smaller impact on projection matrices than others. Moreover, the three worst calibration sensitivity measures are achieved for the images for which the calibration process is performed with all cube vertices except the one which is least distant to the origin of the scene.

Considering the distances of those particular calibration points from the origin of the 3D scene coordinate system, a general tendency can be observed. The points lying furthest from the origin have the greatest impact on the projection matrix entries, in other words, calibration is more sensitive to the points lying

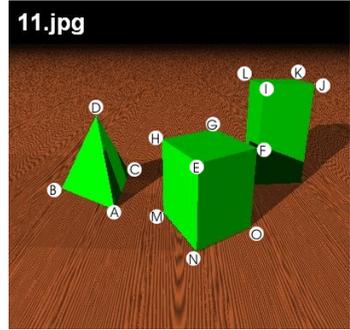


Fig. 2. Image 11.jpg and vertex labels

Table 1. Calibration sensitivity to individual calibration points and combined sensitivity to all calibration points (multiplied by 10^4)

Image \ Point	E	F	G	H	M	N	O	P	Combined
01.jpg	2	3	21	1	-	5	1	5	22.5
02.jpg	31	92	19	25	17	50	17	-	116.15
03.jpg	21	3	6	14	-	10	17	2	32.8
04.jpg	16	2	0	5	4	-	3	1	17.63
05.jpg	12	1	0	4	4	-	5	1	14.25
06.jpg	2	1	0	1	1	-	0	0	2.65
07.jpg	7	6	1	1	2	6	-	3	12.04
08.jpg	1	2	2	2	0	0	-	2	4.12
09.jpg	0	13	1	8	9	5	-	1	18.47
10.jpg	18	10	3	50	5	24	10	-	60.28
11.jpg	288	115	8	27	2	112	35	-	332.8

further from the origin. A possible reason for this is that the absolute values of coordinates are multipliers in the formulas for projection matrix derivatives. Therefore, if they are smaller, they lead to smaller derivatives. Indeed, translating the system of 3D coordinates so that the center of the base of the cube was close to the origin resulted in a calibration error that was a fraction of its previous value, i.e. the projection matrix entries appeared to be less sensitive to input errors. Conversely, setting the system of 3D coordinates so that the cube was standing further from the origin resulted in a calibration error that was a multiple of its original value.

A more detailed examination of the results revealed that for almost all images and points the entries $P_{1,4}$ and $P_{2,4}$ are the most sensitive to errors in the $x^{(k)}$ and $y^{(k)}$ coordinate.

The tests can however only serve as an illustration of the observed tendency and cannot be treated as a proof for the hypothesis. Even if the tendency is considered as a rule, one can ask whether a small sensitivity of the projection matrix to calibration inputs is desirable or not. Moreover, if the calibration points are located far from the camera, a small difference in 2D image coordinates results in a large difference in 3D scene coordinates. This especially applies to picture 06.jpg, where the distance between the cube vertices used for calibration and the camera is definitely the greatest among all the images. From the presented point of view, a small sensitivity to input errors works against the quality of the process in the sense of precise reconstruction. This feature of camera calibration should then be taken into account, when choosing the camera position for best reconstruction results.

3D reconstruction was performed for every pair among the 11 views, i.e. for a total of 55 pairs. In general, 8 feature points - the vertices of the tetrahedron and cuboid - were subject to 3D reconstruction. However, due to the fact that not all of these points are visible in all pictures, the number of reconstructed points varied from five to eight. Each pair of images was used for reconstruction and yielded data which we used to evaluate error propagation.

We begin the analysis of results by comparing the reconstructed 3D coordinates with the ideal ones taken from POVRay. In order to measure the quality of each 3D reconstruction, we calculated the mean-square error of all 3D coordinates reconstructed for a given pair. The results are shown in Table 2.

Table 2. The quality of 3D reconstruction

Image	01.jpg	02.jpg	03.jpg	04.jpg	05.jpg	06.jpg	07.jpg	08.jpg	09.jpg	10.jpg	11.jpg
01.jpg		0.0018	0.0017	0.0101	0.0022	0.0202	0.0025	0.0062	0.0025	0.0018	0.0016
02.jpg	0.0018		0.0004	0.0021	0.0013	0.0286	0.0020	0.0007	0.0019	0.0011	0.0104
03.jpg	0.0017	0.0004		0.0017	0.0005	0.0098	0.0018	0.0005	0.0013	0.0002	0.0009
04.jpg	0.0101	0.0021	0.0017		1.2742	0.2662	0.0041	0.0011	0.0015	0.0010	0.0013
05.jpg	0.0022	0.0013	0.0005	1.2742		0.0867	0.0031	0.0005	0.0018	0.0003	0.0005
06.jpg	0.0202	0.0286	0.0098	0.2662	0.0867		0.0219	0.0133	0.0160	0.0114	0.0198
07.jpg	0.0025	0.0020	0.0018	0.0041	0.0031	0.0219		0.0115	0.0026	0.0021	0.0020
08.jpg	0.0062	0.0007	0.0005	0.0011	0.0005	0.0133	0.0115		0.0127	0.0001	0.0007
09.jpg	0.0025	0.0019	0.0013	0.0015	0.0018	0.0160	0.0026	0.0127		0.0058	0.0016
10.jpg	0.0018	0.0011	0.0002	0.0010	0.0003	0.0114	0.0021	0.0001	0.0058		0.0017
11.jpg	0.0016	0.0104	0.0009	0.0013	0.0005	0.0198	0.0020	0.0007	0.0016	0.0017	

Two facts can be observed. First, the pair of images taken from points of view that differed least (04.jpg and 05.jpg) resulted in the least precise reconstruction. The reason for this is that narrowly spaced viewpoints lead to small angles between gaze directions. Therefore, a slight inaccuracy in image coordinates results in a large change of reconstructed 3D location, especially the depth. In the extreme case of coincident cameras, 3D reconstruction is impossible.

The theoretically calculated sensitivity of reconstructed points to errors in the input 2D coordinates - according to formula (16) - is also highest for this image pair.

The second observation regarding the quality of reconstruction with use of picture 06.jpg, the one with the projection matrix least sensitive to input data errors, is quite surprising. This image yields the worst performance (in terms of overall error in pairs with every other image). This shows that a small sensitivity of the estimated projection matrix to input data errors does not guarantee a good reconstruction. We should emphasize that while talking about calibration precision, we mean the precision of the estimation of the projection matrix and disregard the impact of its entries on the reconstruction quality.

All the 11 projection matrices obtained in the calibration stage were examined regarding the sensitivity of their entries to input 2D coordinates of the calibration points. The most sensitive entry was identified for every image, feature point, and coordinate. In 133 cases out of 154, this was either $P_{1,4}$ or $P_{2,4}$. Most of the rest have single deviations from that rule and there are only two projection matrices having more than two. A question arises how this influences the quality of reconstruction.

We also considered the accuracy of 3D reconstruction of individual points. We found that the points most distant from origin were reconstructed less accurately. Using the differential method as above, the sensitivity of these reconstructed 3D points (points J and K on Fig. 1) to errors in the 2D inputs was determined for all reconstruction image pairs. Pairs including picture 06.jpg turned out to be slightly more sensitive to the errors than others. The same was true, to a lesser extent, about reconstructed points other than J and K.

Our next experiment used an analytical model of camera pairs and a scene.

The scene consisted of a cube and tetrahedron, each with an edge length of 13 cm, both centered at the origin of the 3D coordinate system. This meant they overlapped, but in a simulated environment this was not a problem.

The intrinsic parameters of the cameras represented a focal distance (35mm equivalent) of 52mm and an image sensor with 3072 x 2048 pixels (for an aspect ratio of 3:2), placed symmetrically with respect to the optical axis.

The extrinsic parameters positioned the camera to look directly at the origin of the 3D coordinates from a distance of 100 cm. 72 such virtual cameras were placed on a horizontal circle (the Y axis being vertical) around the origin.

Each camera was virtually calibrated using seven of the vertices of the cube, imitating a real scene where it is impossible to see all 8. The first camera was then paired with every other one, resulting in 71 pairs. For each pair, the 3D position of the vertices of the tetrahedron were reconstructed from their coordinates projected by the two cameras. The camera parameters and the simulation results were substituted to the formulas of Sections 2.2 and 2.3 to compute the sensitivities of reconstructed 3D points to errors in the 2D coordinates used for either calibration or reconstruction. Results are presented in Figure 3.

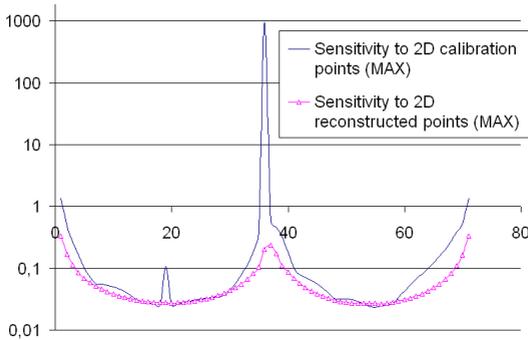


Fig. 3. Sensitivity of reconstruction as function of choice of second camera

The horizontal axis is the number (2 through 72) of the camera forming a pair with Camera 1. The central point (camera 37) represents a pair of cameras facing each other across the scene. The sensitivity on the vertical axis is expressed in centimeters of reconstruction error per pixel of error in image coordinate.

It can be seen that the sensitivity increases dramatically when the optical axes of the two cameras are close to each other, whether gazing in the same or opposite directions. There is also an unexpected, smaller increase in sensitivity to 2D coordinate error in calibration, occurring for perpendicular cameras. In this position, the optical axes of both cameras are parallel to edges of the calibration cube, which may have affected the stability of our matrix computations. Further experiments will be needed to clarify this.

To summarize: the experiments and theoretical analysis have brought some valuable observations, which may serve as material for further and more detailed discussion of the factors that condition the quality of the whole stereovision process, as well as its particular stages. The main observations are:

- The choice of calibration points influences the precision of the recovered projection matrix. Points closer to the origin of the assumed system of 3D coordinates have a smaller impact on the projection matrix entries.
- Increasing the distance between the camera and the scene increases the impact of 2D reconstruction input errors on the quality of the process.
- Bringing the two cameras closer together, or facing each other, decreases the 3D reconstruction quality and increases the impact of 2D reconstruction input errors on the quality of the process.

Acknowledgments. This work was supported in part by the Polish Ministry of Science and Higher Education, under the research grant N N516 1862 33.

References

1. Blostein, S., Huang, T.: Error analysis in stereo determination of 3-d point positions. *IEEE Trans. Pat. An. Mach. Int. PAMI-9(6)* (1987)
2. Grossmann, E., Santos-Victor, J.: Least-squares 3D reconstruction from one or more views and geometric clues. *Comp. Vis. Im. Und.* 99, 151–174 (2005)
3. Hartley, R., Zisserman, A.: *Multiple view geometry in computer vision*. Cambridge University Press, Cambridge (2003)
4. Kamberova, G., Bajcsy, R.: Precision in 3-D points reconstructed from stereo (1997), <http://www.cis.upenn.edu/~kamberov/doc/eccv.ps.gz>
5. Ma, Y., Soatto, S., Košecák, J., Sastry, S.: An invitation to 3-D vision. In: *IAM*, vol. 26. Springer, NY (2004)
6. Min, S., Rixin, H., Daojun, W.: Precision analysis to 3D reconstruction from image sequences
7. Papadopoulo, T., Lourakis, M.: Estimating the Jacobian of the singular value decomposition: Theory and applications, INRIA RR 3961 (2000)
8. Solina, F.: Errors in stereo due to quantization, Univ. Pennsylvania, Tech. Rep. MS-CIS-85-34 (1985)
9. Telle, B., Stasse, O., Yokoi, K., Ueshiba, T., Tomita, F.: Three characterisations of 3D reconstruction uncertainty with bounded error. *Proc. IEEE*, 3894–3899 (2005)
10. Verri, A., Torre, V.: Absolute depth estimates in stereopsis. *J. Opt. Soc. Amer.* 3(3), 297–299 (1986)