Robust Pose Estimation from a Planar Target

Gerald Schweighofer and Axel Pinz, *Member*, *IEEE*

Abstract—In theory, the pose of a calibrated camera can be uniquely determined from a minimum of four coplanar but noncollinear points. In practice, there are many applications of camera pose tracking from planar targets and there is also a number of recent pose estimation algorithms which perform this task in real-time, but all of these algorithms suffer from pose ambiguities. This paper investigates the pose ambiguity for planar targets viewed by a perspective camera. We show that pose ambiguities—two distinct local minima of the according error function—exist even for cases with wide angle lenses and close range targets. We give a comprehensive interpretation of the two minima and derive an analytical solution that locates the second minimum. Based on this solution, we develop a new algorithm for unique and robust pose estimation from a planar target. In the experimental evaluation, this algorithm outperforms four state-of-the-art pose estimation algorithms.

Index Terms—Camera pose ambiguity, pose tracking.

1 Introduction

There are many applications of pose estimation, where 6 degrees of freedom of a camera's pose have to be calculated from known correspondences with known scene structure. This can be done from a single image or from an image sequence. In photogrammetry, this problem is known as space resectioning and it is often solved offline by bundle adjustment techniques, achieving very high precision and, at the same time, high robustness against outliers. In general, there are several ways to solve this problem, as long as many points can be used and when the pose can be calculated offline (e.g., taking information from frames $n+k$ into account to calculate the pose for frame n). Online pose tracking requires calculating the camera pose for each frame, in real-time. Often, interest points are extracted from a target and the camera pose is calculated relative to the target's pose in the scene. In theory, pose can be calculated from four or more coplanar but noncollinear points, if the intrinsic parameters of the camera are known; this remains a critical configuration for an uncalibrated camera, even if many coplanar points are available [1].

In the computer vision literature, several approaches to pose estimation are known. Most of them work for arbitrary 3D target point configurations [2], [3], [4], some have been extended to use points and lines [5], [6], and some work also for coplanar points [3], [4]. Recent success has also been reported for online structure *and* motion estimation [7]; where many interest points are extracted, frame-to-frame correspondence is rather easy and no a priori reference to a scene coordinate system is required (for early work in this direction see [8]).

Augmented Reality (AR) is a main area of application of vision-based tracking systems which are based on planar targets. Examples of such systems include the widely used ARToolkit [9] and the system of Malik et al. [10] (they claim to be more precise than ARToolkit due to an improved target design). Kawano et al. [11] discuss a number of further planar markers for AR and present their own coded planar target. Users of such systems observe that vision-based pose is not very precise, which results in significant jitter, and not very robust, suffering from pose jumps and gross pose outliers. We also provide experimental evidence in this paper, where we compare several state-of-the-art pose algorithms and observe severe pose jumps, which should not occur.

These pose ambiguities have also been discussed by Oberkampf et al. [12]. They give a straightforward interpretation for the case of orthographic projection and they develop their POSIT algorithm for planar targets, which uses scaled orthographic projection at each iteration step. POSIT starts from the two minima under orthography, maintains two alternative solutions, and, finally, decides for the better one based on a distance measure E, which is calculated from reprojection errors in the image space. However, this approach does not analyze the perspective situation, but rather provides an efficient heuristic to deal with the two minima.

This paper tackles the general case of perspective projection. We show that pose ambiguities exist also for cases with wide angle lenses and close range targets. We give a comprehensive interpretation of these ambiguities and develop a new algorithm for a unique and robust solution to pose estimation from a planar target. We start with a simple geometric interpretation of pose ambiguity and show, in an illustrative example, how two local minima of an according error function can develop (Section 2). Section 3 presents an algorithm that analyzes the error function numerically and provides the actual number of local minima for a given experimental configuration. Based on our experimental evidence, that there are, in general, two local minima, we develop our new robust pose estimation algorithm in Section 4. Section 5 presents experimental results and comparison with state-of-the-art pose algorithms, and conclusions are drawn in Section 6.

2 Pose Ambiguity/Geometric Interpretation

Camera pose estimation is discussed for a calibrated camera (with known interior parameters) as the problem of finding the six exterior parameters of the camera: orientation $R = f(\alpha, \beta, \gamma)$ and position $\mathbf{t} = [t_x, t_y, t_z]^T$ of the camera with respect to a scene coordinate system. Fig. 1 shows the center of projection C_C (camera center, origin of the camera coordinate system), the image plane, and a planar model (model points P_i in the plane Π). Without loss of generality, we assume that the model coordinate system is located in the model center C_M and coincides with the origin of the scene coordinate system. Thus, camera position $\mathbf{t}$ is represented by the vector from C_C to C_M and camera orientation can be represented as rotations of the model plane, as shown for a rotation α around the y-axis of the model coordinate system.

We assume n coplanar model points $\mathbf{p}_i = [p_{i_x}\ p_{i_y}\ 0]^T$ in scene coordinates which are transformed to camera coordinates $\mathbf{v}_i$ by

$$\mathbf{v}_i \propto R\mathbf{p}_i + \mathbf{t}, \qquad (1)$$

where $\propto$ denotes "directly proportional" because $\mathbf{v}_i$ are measured up to an unknown scale factor. We further on refer to $\mathbf{v}_i$ as points in camera coordinates and to $\hat{\mathbf{v}}_i$ as their measurements in the image (also in camera coordinates, but imprecise due to noise). A *pose estimation* algorithm has to find values for $\hat{R}$ and $\hat{t}$ that minimize an error function. In principle, there are two possible choices: *image space error* (used in bundle-adjustment and by [2], [12])

$$E_{is}(\hat{R}, \hat{\mathbf{t}}) = \sum_{i=1}^{n} \left[\left(\frac{\hat{v}_{ix}}{\hat{v}_{iz}} - \frac{R_x^t \mathbf{p}_i + t_x}{R_z^t \mathbf{p}_i + t_z} \right)^2 + \left(\frac{\hat{v}_{iy}}{\hat{v}_{iz}} - \frac{R_y^t \mathbf{p}_i + t_y}{R_z^t \mathbf{p}_i + t_z} \right)^2 \right],$$
$$R = \begin{bmatrix} R_x^T \\ R_y^T \\ R_z^T \end{bmatrix} \qquad (2)$$

and *object-space error*, as used by [3], [6],

$$E_{os}(\hat{R}, \hat{\mathbf{t}}) = \sum_{i=1}^{n} \left\| \left(I - \hat{V}_i \right) \left(\hat{R}\mathbf{p}_i + \hat{\mathbf{t}} \right) \right\|^2 \quad \text{with} \quad \hat{V}_i = \frac{\hat{\mathbf{v}}_i \hat{\mathbf{v}}_i^t}{\hat{\mathbf{v}}_i^t \hat{\mathbf{v}}_i}. \qquad (3)$$

- *The authors are with the Institute of Electrical Measurement and Measurement Signal Processing, Graz University of Technology, Kopernikusgasse 24/IV, 8010 Graz, Austria.*
E-mail: {gerald.schweighofer, axel.pinz}@tugraz.at.

Manuscript received 13 May 2005; revised 1 June 2006; accepted 5 June 2006; published online 12 Oct. 2006.
Recommended for acceptance by M. Pollefeys.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0247-0505.

0162-8828/06/$20.00 © 2006 IEEE Published by the IEEE Computer Society

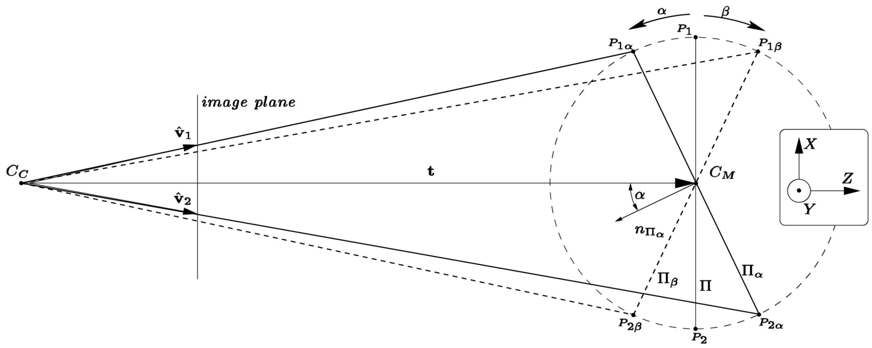
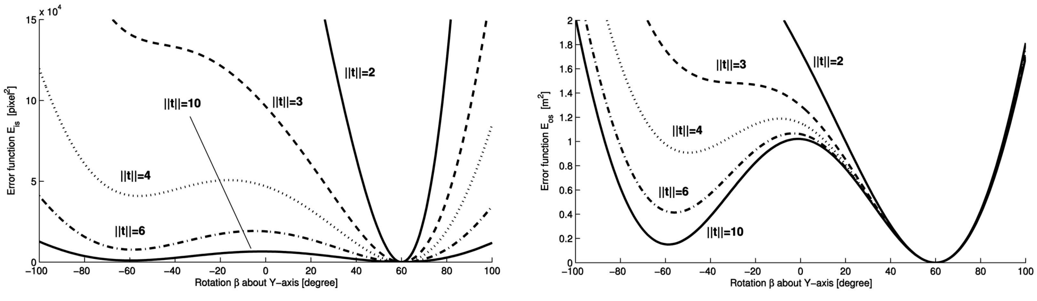
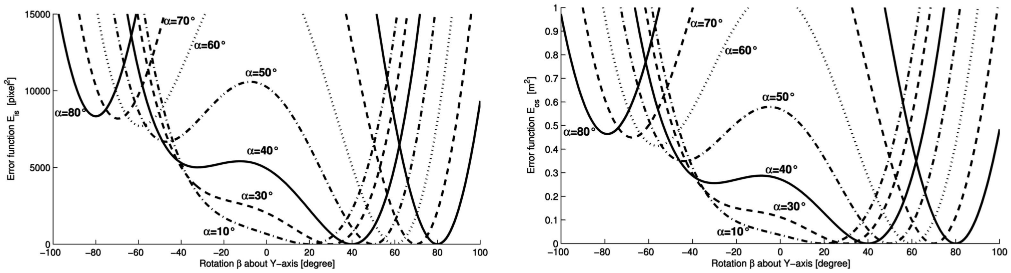


Fig. 1. Pose Ambiguities with perspective projection.



(a) (b)

Fig. 2. (a) E_{is} and (b) E_{os} for $\alpha = 60°$ and varying distances $\|\mathbf{t}\|$.



(a) (b)

Fig. 3. (a) E_{is} and (b) E_{os} for various rotation angles α.

Pose ambiguity denotes situations where E_{is} or E_{os} have several local minima for a given configuration. We now illustrate pose ambiguity for a comprehensive example. We assume a distinct rotation α around the y-axis, as depicted in Fig. 1, and point out that there may exist a second, different angle β which also leads to a local minimum of the according error function E_{is} or E_{os}. Since the existence of such a minimum depends on all parameters of (2) or (3), we subsequently assume certain parameter configurations and just vary one parameter at a time ($\|\mathbf{t}\|$ or α).[1] Fig. 2 shows how the error functions change for $\alpha = 60°$ and zero noise ($\hat{\mathbf{v}}_i = \mathbf{v}_i$), depending on different distances $\|\mathbf{t}\|$ between model center C_M and camera center C_C. In all cases, there is one absolute minimum at $\alpha = 60°$ where $E_{is} = E_{os} = 0$. For $\|\mathbf{t}\| = 4m, 6m, 10m$, we see local minima of increasing significance (decreasing E_{os}) at $\beta = -49.6°$, $-56.0°$, and $-58.6°$. With increasing distance $\|\mathbf{t}\|$ between model and camera, the effect of perspective projection decreases. For $\|\mathbf{t}\| = \infty$, we would have *orthographic projection* with a second minimum of $E_{is} = E_{os} = 0$ at $\beta = -60°$. For $\|\mathbf{t}\| < 3.06m$ ($\approx 36.20°$ field of view of the target), there exists only one minimum of the error function ($E_{is} = E_{os} = 0$ at $\alpha = 60°$). Fig. 3 depicts the situation for constant $\|\mathbf{t}\| = 6m$ and varying rotation angles α. Again, we assume a noise-free case and obtain $E_{is} = E_{os} = 0$ for each α. For $\alpha = 40°, 50°, 60°, 70°$, and $80°$, we see a second local minimum of E_{os} at $\beta = -30.1°$, $-43.9°$, $-56.0°$, $-76.5°$, and $-78.8°$. For $\alpha < 34.8°$, there exists no second local minimum of E_{os}.

This very simple example already reveals that there *is* pose ambiguity, not only under orthography, but also for close range perspective, especially when the model plane is significantly tilted. We also note that the magnitudes and the positions of local minima are very similar for the two error functions E_{is} and E_{os}. In our example, we could observe the development of a second prominent minimum and the correct minimum would always be found for zero error functions because no measurement noise was assumed. In the general case, we will have measurement noise ($\hat{\mathbf{v}}_i \neq \mathbf{v}_i$), and the number of local minima will depend on *all* parameters of E_{is} or E_{os}. In the remainder of this paper, we derive results for E_{os}, which has also been used by [3], [6] and is easier to parameterize than E_{is}. For general configurations (arbitrary number and positions of points in the model plane, significant measurement noise, rotations about three axes), we first examine ways to estimate the number of local minima in Section 3, and

[1] Our model is a square centered around C_M with $p1 = [1\,1\,0]^T m$, $p2 = [1\,-1\,0]^T m$, $p3 = [-1\,-1\,0]^T m$, $p4 = [-1\,1\,0]^T m$, focal length of the camera is $800\,pixels$, and $pixelsize = 8 \times 10^{-6} m$.

proceed with the development of a robust pose estimation algorithm in Section 4.

3 Local Minima

In this section, we develop a numerical algorithm which finds all minima of E_{os}. From [3], we know that the optimal translation is given by

$$\hat{\mathbf{t}}_{opt} = \frac{1}{n}\left(I - \frac{1}{n}\sum_j \hat{V}_j\right)^{-1} \sum_j (\hat{V}_j - I)\hat{R}\mathbf{p}_j = G\sum_j (\hat{V}_j - I)\hat{R}\mathbf{p}_j, \quad (4)$$

which results in an error function depending only on the rotation $\hat{R}$

$$E_{os}(\hat{R}) = \sum_{i=1}^{n} \left\| (I - \hat{V}_i)\left(\hat{R}\mathbf{p}_i + G\sum_j (\hat{V}_j - I)\hat{R}\mathbf{p}_j\right)\right\|^2. \quad (5)$$

Using Euler angles to parameterize the rotation $\hat{R}$ with the three parameters α, β, and γ,

$$\hat{R}(\alpha, \beta, \gamma) = R_x(\alpha)R_y(\beta)R_z(\gamma), \quad (6)$$

where $R_i(\phi)$ describes a rotation of ϕ degrees about axis i and, using the substitution,

$$\phi_t = \begin{cases} k_\phi \frac{\sqrt{1-\cos(\phi)}}{1+\cos(\phi)} & \text{if } 2k - \frac{\pi}{2} < \phi \le 2k + \frac{\pi}{2} \\ k_\phi \frac{\sqrt{1-\cos(\phi-\pi)}}{1+\cos(\phi-\pi)} & \text{otherwise,} \end{cases} \quad (7)$$
$$k_\phi = \begin{cases} +1 & \text{if } k < \phi \le k + \frac{\pi}{2} \\ -1 & \text{otherwise,} \end{cases}$$

where $k = \ldots, -2\pi, -\pi, 0, \pi, 2\pi, \ldots$, we can write the the trigonometric functions as fraction of polynomials in ϕ_t

$$\begin{matrix} \cos(\phi) = s_\phi \frac{1-\phi_t^2}{1+\phi_t^2} \\ \sin(\phi) = s_\phi \frac{2\phi_t}{1+\phi_t^2} \end{matrix} \quad \text{for } \phi = \alpha, \beta, \gamma \text{ and } s_\phi \in \{-1, 1\}. \quad (8)$$

For example, the rotation $R_x(\alpha)$ about the x-axis with $\alpha \in [0, 2\pi]$ can be written as

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix}$$
$$\leadsto R_{x,s_\alpha}(\alpha_t) = \frac{1}{1+{\alpha_t}^2}\begin{bmatrix} 1+{\alpha_t}^2 & 0 & 0 \\ 0 & s_\alpha(1-\alpha_t^2) & s_\alpha(-2\alpha_t) \\ 0 & s_\alpha(2\alpha_t) & s_\alpha(1-\alpha_t^2) \end{bmatrix}, \quad (9)$$

with two distinct ($s_\alpha = -1$ and $s_\alpha = +1$) functions $R_{x,s_\alpha}(\alpha_t)$ in the interval $\alpha_t \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$. Repeating this process for $R_y(\beta) \leadsto R_{y,s_\beta}(\beta_t)$ and $R_z(\gamma) \leadsto R_{z,s_\gamma}(\gamma_t)$ and considering the symmetry of the rotation ($\hat{R}(\alpha, \beta, \gamma) \equiv \hat{R}(\alpha + \pi, 3\pi - \beta, \gamma + \pi)$), we obtain a piecewise (four different combinations of s_α, s_β, and constant $s_\gamma = 1$) representation of the rotation $\hat{R}(\alpha, \beta, \gamma) \leadsto \hat{R}_{s_\alpha,s_\beta}(\alpha_t, \beta_t, \gamma_t)$. Using this parameterization of the rotation in the error function (5), we get

$$E_{s_\alpha,s_\beta}(\alpha_t, \beta_t, \gamma_t) = \frac{P_{s_\alpha,s_\beta}(\alpha_t, \beta_t, \gamma_t)}{(1+\alpha_t^2)^2(1+\beta_t^2)^2(1+\gamma_t^2)^2}, \quad (10)$$

where $P_{s_\alpha,s_\beta}(\alpha_t, \beta_t, \gamma_t)$ is a multivariate polynomial of order five in the variables α_t, β_t, and γ_t. The coefficients can be estimated from the given model points $\mathbf{p}_i$ and the measured camera coordinates $\hat{\mathbf{v}}_i$. This means that we can replace the original error function (5) with its trigonometric terms by four different functions E_{s_α,s_β}, which are fractions of polynomials. We want to mention explicitly that the original problem (5) is identical to the transformed one (10). Thus, both functions will have the same number of minima, which can be found as roots of the first derivatives

$$\frac{\partial E_{s_\alpha,s_\beta}(\alpha_t, \beta_t, \gamma_t)}{\partial \phi_t} = \frac{\frac{\partial P_{s_\alpha,s_\beta}(\alpha_t,\beta_t,\gamma_t)}{\partial \phi_t}(1+\phi_t^2) - 4\phi_t P_{s_\alpha,s_\beta}(\alpha_t, \beta_t, \gamma_t)}{(1+\phi_t^2)(1+\alpha_t^2)^2(1+\beta_t^2)^2(1+\gamma_t^2)^2} = 0 \quad (11)$$

$$\leadsto E_{\phi_t} = \frac{\partial P_{s_\alpha,s_\beta}(\alpha_t, \beta_t, \gamma_t)}{\partial \phi_t}(1+\phi_t^2) - 4\phi_t P_{s_\alpha,s_\beta}(\alpha_t, \beta_t, \gamma_t) = 0$$
$$\text{for } \phi_t = \alpha_t, \beta_t, \gamma_t. \quad (12)$$

Solutions to the three equations are found by using an interval-based splitting approach [13]. Starting from intervals $\alpha_t \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$, $\beta_t \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$, and $\gamma_t \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$, we estimate the bounds of the three derivatives E_{α_t}, E_{β_t}, and E_{γ_t} using Bernstein expansions (for details, the reader is refered to [13]). An interval will contain one or more stationary points, if all lower bounds of the three functions are negative and all upper bounds are positive. In this case, the interval is divided into two subintervals and the process is repeated until a given size of the intervals (we chose a maximum number of divisions of 20 for each variable, leading to an interval size of $\frac{1}{2^{20}} \approx 2 \times 10^{-6} rad$) is reached.

After this process, we end up with a list of narrow intervals, where the error function may attain a minimum. To check the type of the stationary point (minimum, maximum, or saddle point), we estimate the Hessian matrix

$$H_{\alpha_t,\beta_t,\gamma_t} = \begin{bmatrix} E_{\alpha_t,\alpha_t} & E_{\alpha_t,\beta_t} & E_{\alpha_t,\gamma_t} \\ E_{\beta_t,\alpha_t} & E_{\beta_t,\beta_t} & E_{\beta_t,\gamma_t} \\ E_{\gamma_t,\alpha_t} & E_{\gamma_t,\beta_t} & E_{\gamma_t,\gamma_t} \end{bmatrix} \quad (13)$$

for each interval. A minimum requires that the three eigenvalues of the Hessian matrix are positive. To estimate these eigenvalues, we have to estimate the roots of the characteristic polynom $\det(H\alpha_t, \beta_t, \gamma_t - \lambda I) = 0$. Because we just have lower and upper bounds of the entries of the Hessian matrix for a narrow interval, the estimation of the eigenvalues is performed using interval analysis [14]. Only if the lower bounds of the three eigenvalues are positive, then the stationary point inside this interval is a minimum. If we cannot specify the type of the stationary point (for different signs of lower and upper bound of an eigenvalue), we need to restart the divide process. Finally, this process delivers a list of intervals which contain minima of (10) and all minima of the four different functions ($s_\alpha \in \{-1, 1\}$ and $s_\beta \in \{-1, 1\}$) of (10) have to be collected to obtain a list of all minima of the error function (5). As a final step, only minima with a valid physical interpretation, i.e., solutions where all the model points $\mathbf{p}_i$ are in front of the camera, are selected.

As will be seen from our experiments in Section 5.1, a thorough analysis of our special case (coplanar points in front of the camera viewed under perspective projection) typically delivers two distinct minima. Sometimes, there is only one (the correct) minimum. But, we found not a single case with more than two local minima of E_{os}.

4 Robust Pose Estimation Algorithm

Our new algorithm is based on the following assumptions: There is either one (the correct) minimum or there are two local minima of E_{os} depending on the actual configuration ($\hat{R}$, $\hat{\mathbf{t}}$, $\mathbf{p}_i$, and $\hat{\mathbf{v}}_i$). For real images with measurement noise ($\mathbf{v}_i \neq \hat{\mathbf{v}}_i$), E_{os} will, in general, be above zero, but in cases of two local minima, the error should be lower for the correct pose. Existing pose estimation algorithms search for a minimum of E_{os} and will often return the wrong pose (up to 50 percent of the cases, as is shown in Section 5).

To develop our algorithm, we first assume a known pose ($\hat{R}_1$, $\hat{\mathbf{t}}_1$), which we can get from any pose estimation algorithm, and then use this first guess of a pose to estimate a second pose, which also minimizes the error function E_{os}. An analytic solution for the second pose can be obtained by modifying the general transformation (1) in a way that, finally, E_{os} only depends on a rotation about the y-axis $R_y(\beta)$ and $\mathbf{t}$. Subsequently, we switch between presentations of general equations ((14), (16), and (19)) and specific calculations (R_x and R_z) based on the initial first pose $(\hat{R}_1, \hat{\mathbf{t}}_1)$.

Assume model points $\mathbf{p}_i$ which are measured in the image as $\hat{\mathbf{v}}_i$ (see (1)) such that

$$\hat{\mathbf{v}}_i \approx \mathbf{v}_i \propto R\mathbf{p}_i + \mathbf{t}. \tag{14}$$

Without loss of generality,[2] we can multiply both sides of (14) with R_t to get a transformed system

$$R_t\hat{\mathbf{v}}_i \approx R_t\mathbf{v}_i \propto R_tR\mathbf{p}_i + R_t\mathbf{t}, \tag{15}$$

such that $R_t\hat{\mathbf{t}}_1 = [0\ 0\ \|\hat{\mathbf{t}}_1\|]^T$. This results in a projection of the model center C_M ($[0\ 0\ 0]^T$) to the image as $[0\ 0\ \|\hat{\mathbf{t}}_1\|]^T$. Let

$$\tilde{\mathbf{v}}_i = R_t\hat{\mathbf{v}}_i \quad \tilde{\mathbf{t}}_1 = R_t\hat{\mathbf{t}}_1 \quad \tilde{R}_1 = R_t\hat{R}_1. \tag{16}$$

The pose $(\tilde{R}_1, \tilde{\mathbf{t}}_1)$ minimizes

$$E_{os}(\tilde{R}, \tilde{\mathbf{t}}) = \sum_{i=1}^{n} \left\| \left(I - \tilde{V}_i\right)\left(\tilde{R}\mathbf{p}_i + \tilde{\mathbf{t}}\right) \right\|^2. \tag{17}$$

Without loss of generality, we introduce a rotation matrix $\tilde{R}_z$, such that

$$E_{os}(\tilde{R}, \tilde{\mathbf{t}}) = \sum_{i=1}^{n} \left\| \left(I - \tilde{V}_i\right)\left(\tilde{R}\underbrace{\tilde{R}_z\tilde{R}_z^{-1}}_{I}\mathbf{p}_i + \tilde{\mathbf{t}}\right) \right\|^2. \tag{18}$$

The rotation $\tilde{R}_z^{-1}$ rotates the planar model $\mathbf{p}_i$, where all $p_{i_z} = 0$, only about the z-axis such that the rotated model $\tilde{\mathbf{p}}_i = \tilde{R}_z^{-1}\mathbf{p}_i$ is also planar with $z = 0$. The rotation matrix $\tilde{R}_1\tilde{R}_z$ can be decomposed[3] into the product of three rotations $\tilde{R}_1\tilde{R}_z = R_z(\tilde{\gamma}_1)R_y(\tilde{\beta}_1)R_x(\tilde{\alpha}_1)$, where $R_i(\phi)$ describes a rotation of ϕ degrees about axis i. By selecting $\tilde{R}_z$ such that $\tilde{\alpha}_1 = 0$, we obtain another transformed system

$$\tilde{\mathbf{v}}_i \approx R_z(\tilde{\gamma})R_y(\tilde{\beta})\tilde{\mathbf{p}}_i + \tilde{\mathbf{t}} \tag{19}$$

with the corresponding error function

$$E_{os}(\tilde{\gamma}, \tilde{\beta}, \tilde{\mathbf{t}}) = \sum_{i=1}^{n} \left\| \left(I - \tilde{V}_i\right)\left(R_z(\tilde{\gamma})R_y(\tilde{\beta})\tilde{\mathbf{p}}_i + \tilde{\mathbf{t}}\right) \right\|^2. \tag{20}$$

Equations (19) and (20) are equal to (15) and (17) in the case of using the known pose $(\tilde{R}_1, \tilde{\mathbf{t}}_1)$ or $\left(f(0, \tilde{\beta}_1, \tilde{\gamma}_1), \tilde{\mathbf{t}}_1\right)$.

From Section 2, we know that there may be two minima of the error function E_{os} with regard to a rotation about one axis (y-axis) depending on the parameters. In our transformed system (19), we have a rotation about this axis (y-axis) and a rotation about the z-axis.

Let us discuss the effect of the rotation $R_z(\tilde{\gamma})$. We know from our normalization step (16) that $\tilde{\mathbf{t}}_1 = [0\,0\,\|\hat{\mathbf{t}}_1\|]^T$. In this case, we can rewrite (19) as

$$\tilde{\mathbf{v}}_i \approx R_z(\tilde{\gamma})(R_y(\tilde{\beta})\tilde{\mathbf{p}}_i + \tilde{\mathbf{t}}_1) \tag{21}$$

because $R_z(\tilde{\gamma})\tilde{\mathbf{t}}_1 = \tilde{\mathbf{t}}_1$. Thus, $R_z(\tilde{\gamma})$ is a rotation just around the optical axis of the camera. This rotation leaves the geometric relation between image plane and model plane invariant and just affects image coordinates. Thus, we can just search for local minima of E_{os} with respect to β.

This reparameterization of E_{os} leads us to our new "*Robust Pose Estimation Algorithm for Planar Targets:*"

1. Estimate a first pose $\hat{P}_1 = (\hat{R}_1, \hat{\mathbf{t}}_1)$ by applying any existing iterative pose estimation algorithm. In our experiments, we used the iterative algorithm proposed by [3]. $\hat{P}_1$ is one local minimum of E_{os}. Our goal is to analytically derive an estimate of the second local minimum, if such a minimum exists.
2. Transform the coordinate system according to (16) to get $\tilde{P}_1 = (\tilde{R}_1, \tilde{\mathbf{t}}_1)$.
3. Estimate $\tilde{R}_z$ as described in (18) and (19) to obtain the transformed system and the parameters of the first pose ($\tilde{\gamma}_1$ and $\tilde{\beta}_1$).
4. Fix $\tilde{\gamma} = \tilde{\gamma}_1$ and estimate all local minima of (20) for the parameters $\tilde{\beta}$ and $\tilde{\mathbf{t}}$. For details, see Section 4.1 below.
5. Undo the transformations of Steps 1 and 2 for all local minima to obtain poses $\hat{P}_i$.
6. Use all poses $\hat{P}_i$ as an initial value for the iterative pose estimation algorithm [3] to get final poses $P_i^\star$.
7. Decide the final and correct pose, which has the lowest error E_{os}.

4.1 Estimation of the Local Minima

We discuss here how to estimate the local minima of (20) for given $\tilde{\mathbf{p}}_i$,$\tilde{\mathbf{v}}_i$, and $R_z(\tilde{\gamma})$. We can solve (20) for the optimal translation $\tilde{\mathbf{t}}_{opt}$ with regard to the minimization of E_{os} by derivating (20) such that [3]

$$\frac{\partial E_{os}}{\partial \tilde{\mathbf{t}}} = 0 \ \Rightarrow\ \tilde{\mathbf{t}}_{opt}(\tilde{\beta}) = G\sum_j \big(\tilde{V}_j - I\big)R_z(\tilde{\gamma})R_y(\tilde{\beta})\tilde{\mathbf{p}}_j \tag{22}$$

$$\text{with } \tilde{G} \ = \ \frac{1}{n}\left(I - \frac{1}{n}\sum_j \tilde{V}_j\right)^{-1}, \tag{23}$$

where $\tilde{G}$ is a constant which depends only on measured image positions $\tilde{\mathbf{v}}_i$.

By plugging $\tilde{\mathbf{t}}_{opt}(R)$ into (20), we obtain an error function which only depends on $\tilde{\beta}$ (the rotation about the y-axis):

$$\begin{aligned} E_{os}(\tilde{\beta}) = \sum_{i=1}^{n} \|(I - \tilde{V}_i)(R_z(\tilde{\gamma})R_y(\tilde{\beta})\tilde{\mathbf{p}}_i \\ + \tilde{G}\sum_{j=1}^{n}(\tilde{V}_j - I)R_z(\tilde{\gamma})R_y(\tilde{\beta})\tilde{\mathbf{p}}_j)\|^2. \end{aligned} \tag{24}$$

Simplifying $R_y(\tilde{\beta})$, by substituting ($\tilde{\beta}_t = \tan\frac{1}{2\tilde{\beta}}$), we obtain

$$\begin{aligned} R_y(\tilde{\beta}) &= \begin{bmatrix} \cos(\tilde{\beta}) & 0 & \sin(\tilde{\beta}) \\ 0 & 1 & 0 \\ -\sin(\tilde{\beta}) & 0 & \cos(\tilde{\beta}) \end{bmatrix}; \\ R_y(\tilde{\beta}_t) &= \frac{1}{1 + \tilde{\beta}_t^{\,2}} \begin{bmatrix} 1 - \tilde{\beta}_t^{\,2} & 0 & 2\,\tilde{\beta}_t \\ 0 & 1 + \tilde{\beta}_t^{\,2} & 0 \\ -2\,\tilde{\beta}_t & 0 & 1 - \tilde{\beta}_t^{\,2} \end{bmatrix}. \end{aligned} \tag{25}$$

By plugging this into (24), we get a function $E_{os}(\tilde{\beta}_t)$ which now only depends on $\tilde{\beta}_t$. To obtain all stationary points of this function, we need to solve

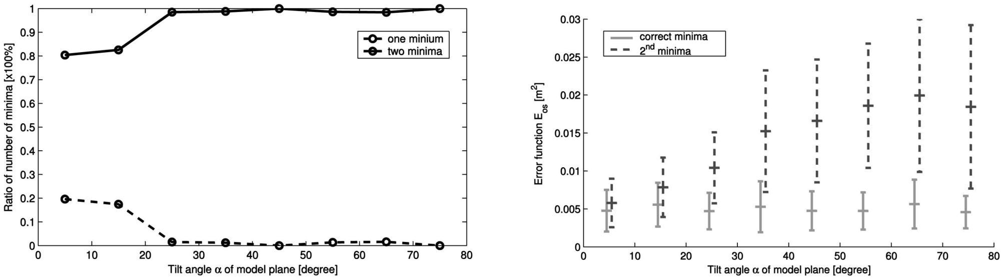
$$\frac{\partial E_{os}(\tilde{\beta}_t)}{\partial \tilde{\beta}_t} = 0, \tag{26}$$

which is a polynomial of degree four and can be easily solved. In general, we will obtain four solutions, which represent those rotations about the y-axis, where E_{os} attains an extremum. There are up to two minima, which are selected as those stationary points with $\frac{\partial^2 E_{os}(\tilde{\beta}_t)}{\tilde{\beta}_t} > 0$.

We want to mention explicitly that the number of solutions (one or two) depends on the pose of the camera (R and $\mathbf{t}$) and on the model points ($\mathbf{p}_i$). For different models, but same pose of the camera, the coefficients of the fourth order polynomial (26) are different.

[2] Note that this transformation does not affect the shape of the error function. Thus, minima of the transformed system will also be minima of the original system. See Appendix A for details.

[3] For details on how to obtain $\tilde{R}_z$, see Appendix B.



(a)

(b)

Fig. 4. Simulation results for the number of minima of E_{os}. (a) Rate of found minima. (b) Mean values and standard deviation for correct and second minima.

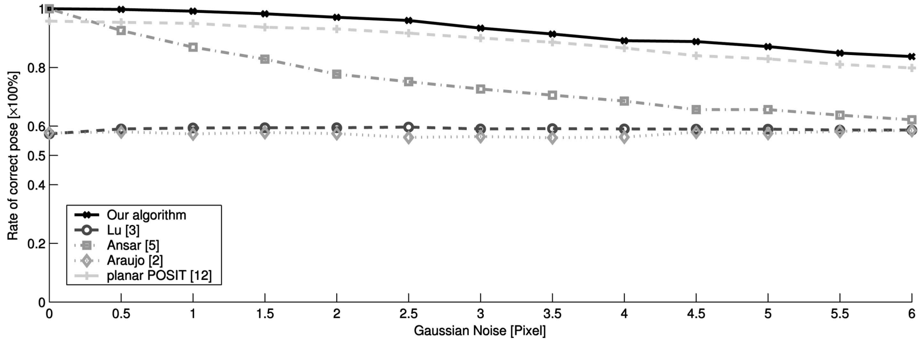


Fig. 5. Rate of choosing the correct pose (Solid line: Our algorithm).

5 Experimental Results

For all experiments, we used the following setup:

1. For each test, we generated a random model consisting of 10 points $p_i = [x_i \; y_i \; z_i]^T$ in the range $x_i = [-1, 1]$, $y_i = [-1, 1]$, and $z_i = 0$ to be planar.
2. For each test, we generated a random rotation R of the pose.
3. The translation vector $\mathbf{t}$ of the pose was chosen such that the measured image points $\mathbf{v}_i$ were located in the image at a random position and with a maximum size of the Feret box of 200 pixels.[4]
4. To each image point $\mathbf{v}_i$, Gaussian noise was added to get $\hat{\mathbf{v}}_i$.

5.1 Estimation of the Number of Minima of E_{os}

We used the algorithm presented in Section 3 to estimate the number of minima for 600 different experiments (with two pixels Gaussian noise). Fig. 4a shows the rates of found minima of this simulation plotted against the tilt angle α of the model plane (see Fig. 1). The rates for tilt angles are averaged (the values for $\alpha = 5$ and 15 average the rates for all angles in [0,9.99], [10,19.99], etc.). First, we want to mention that, in none of our experiments, we get more than two minima. We further see that the rate of appearance of two minima decreases with low angles α, but on average there are at least 80 percent of cases with two minima. This is similar to our geometric interpretation (see Fig. 3). Fig. 4b shows the average Error-values (mean and standard deviation) for these two minima. E_{os} is always lower for the correct minima, but for small tilt angles α, they are very similar. The average number of intervals that had to be inspected to get the final list of minima was approximately 60.000, which, in turn, required about 1 hour of computation time per experiment on a standard 2.4GHz PC.

5.2 Robust Pose Estimation Algorithm

To test the proposed robust pose estimation algorithm (Section 4), we generated 1,000 different models and poses for 13 different noise levels reaching from zero to 6 pixels. In Fig. 5, the results of our algorithm are compared to four state-of-the-art algorithms. The solid line shows the results of our algorithm, which achieves a rate of 100 percent to find the correct pose in the noise-free case $(\text{Gaussian noise} = 0 \; pixels)$. With increasing noise, the rate decreases down to 83.7 percent for 6 *pixels* Gaussian noise. Results for existing common iterative pose estimation algorithms ([2], [3]) are also given. Because there are most of the time two local minima of the error function, the rate of finding the correct solution with these algorithms is just above 50 percent. Only the POSIT algorithm for coplanar feature points [12] maintains two solutions during its iterations and performs comparable to our algorithm. The slightly less robust results of POSIT can be explained in two ways: Either there are cases where POSIT finds only one minimum in the initialization phase, or the scaled orthographic projection approximates the true perspective case such that, in cases of very similar error functions, the wrong minimum is selected. The algorithm of Ansar and Daniilidis [5] starts at 100 percent for zero noise because this algorithm calculates a direct solution, but its performance also decreases significantly with increasing noise level.

In Step 6 of our algorithm, we use all poses $\hat{P}_i$ as initial values for the iterative algorithm by Lu et al. [3]. Fig. 6 shows the average improvement in terms of rotations from pose $\hat{P}_i$ to the refined pose $P_i^\star$. This can also be interpreted as a measure of mean error of our direct solutions obtained by solving (26). The diagram shows mean and standard deviation for all different noise levels. The mean error is always below three degrees and increases with increasing noise level. From Section 2, we know that the number of the minima of the error function increases with increasing tilt angle α of the model plane. Our experiments show the same behavior. The dashed line in Fig. 7 represents the rate of having only one minimum for the noise-free case. The solid line shows this rate for the case of 2 pixels Gaussian noise. There is either one minimum, or there are exactly two minima. For zero noise, we find one unique solution if the image plane is nearly parallel to the model plane ($\alpha \leq 15^\circ$), but still more than 35 percent of two solutions for 2 pixels Gaussian noise. For $\alpha > 20^\circ$, there are almost always two solutions.

Fig. 8 compares the rates of choosing the correct solution for our algorithm (solid line) with other common pose estimation algorithms ([2], [3], [5], [12]) plotted against the angle α. Here, we used only tests where the Gaussian noise was 2 *pixels*. As we know already from Fig. 7, there is only one solution in the case of small α, so the rate of choosing the correct solution is one for all

[4] Interior parameters: $f_x = f_y = 800$, $x_0 = 320$, and $y_0 = 240$. Image Size: 640×480.

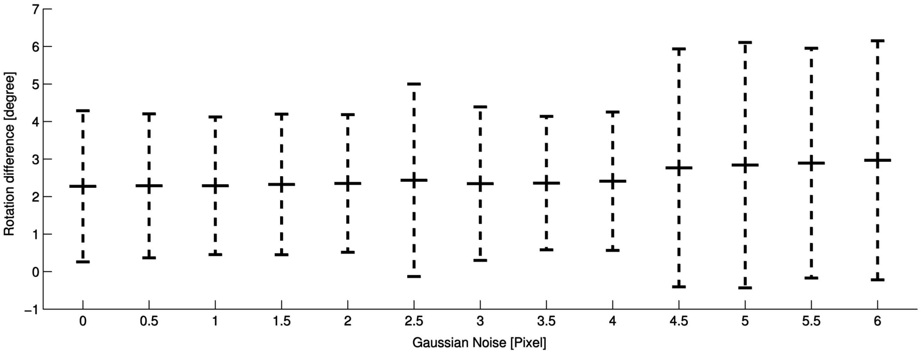


Fig. 6. Average error and standard deviation for $\hat{P}_i - P_i^\star$.

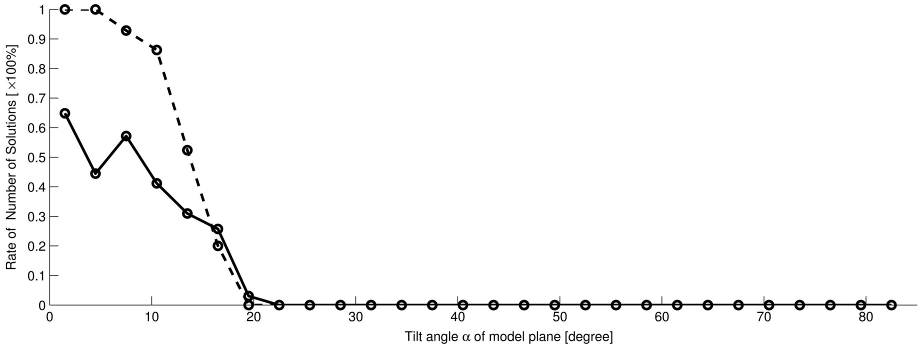


Fig. 7. Rate of having one solution versus angle α (Dashed line: Noise-free case. Solid line: $noise = 2\ pixels$).

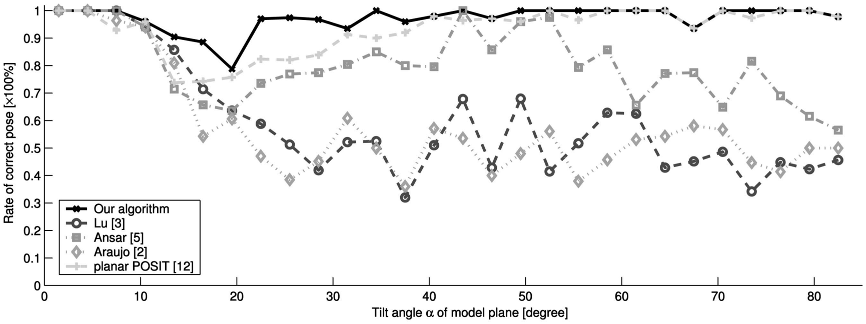


Fig. 8. Rate of correct pose versus angle α for $noise = 2\ pixels$ (Solid line: our algorithm).

algorithms. In cases where two solutions occur, the performance of other algorithms drops down to rates as low as 50 percent to find the correct solution. Our algorithm has a rate of above 95 percent to estimate the correct solution for all angles α except in the range from 10 to 20 degrees. This effect is caused by noise that influences the results in this range of transition from one to two minima (see the overlapping area of solid and dashed lines in Fig. 7). Thus, our algorithm performs slightly better than POSIT [12], but significantly outperforms all other algorithms [2], [3], [5].

6 Conclusions

Except for the work by Oberkampf et al. [12], ambiguities in pose estimation from planar targets have been neglected in the literature. We have presented a thorough analysis of the perspective case and could show that, in general, there exist two local minima of the according error function. These two minima are the reason why pose jumps are observed in many pose tracking applications. Based on these findings, we presented a new "Robust Pose Estimation Algorithm for Planar Targets" which takes the two minima into account to give a robust pose. Our algorithm is based on an analytic solution that locates the second minimum, which makes it more robust against pose jumps than all previous pose estimation algorithms. This new algorithm should be relevant for many applications in AR and navigation.[5]

Appendix A

We show that the error function (3) is invariant to rotation. Multiplying both sides of (1) with a rotation $\mathcal{R}$ gives: $\mathcal{R}\mathbf{v}_i \propto \mathcal{R}R\mathbf{p}_i + \mathcal{R}\mathbf{t}$. The corresponding error function is

$$\begin{aligned}
&\sum_{i=1}^{n}\left\|\left(I-\frac{\mathcal{R}\hat{\mathbf{v}}_i\hat{\mathbf{v}}_i^t\mathcal{R}^t}{\hat{\mathbf{v}}_i^t\mathcal{R}^t\mathcal{R}\hat{\mathbf{v}}_i}\right)(\mathcal{R}\hat{R}\mathbf{p}_i+\mathcal{R}\hat{\mathbf{t}})\right\|^2\\
&=\sum_{i=1}^{n}\left\|\left(\mathcal{R}-\mathcal{R}\frac{\hat{\mathbf{v}}_i\hat{\mathbf{v}}_i^t}{\hat{\mathbf{v}}_i^t\hat{\mathbf{v}}_i}\right)\mathcal{R}^t(\mathcal{R}\hat{R}\mathbf{p}_i+\mathcal{R}\hat{\mathbf{t}})\right\|^2\\
&=\sum_{i=1}^{n}\left\|\mathcal{R}\left(I-\frac{\hat{\mathbf{v}}_i\hat{\mathbf{v}}_i^t}{\hat{\mathbf{v}}_i^t\hat{\mathbf{v}}_i}\right)\mathcal{R}^t(\mathcal{R}\hat{R}\mathbf{p}_i+\mathcal{R}\hat{\mathbf{t}})\right\|^2\\
&=\sum_{i=1}^{n}\left\|\mathcal{R}\left[\left(I-\frac{\hat{\mathbf{v}}_i\hat{\mathbf{v}}_i^t}{\hat{\mathbf{v}}_i^t\hat{\mathbf{v}}_i}\right)(\hat{R}\mathbf{p}_i+\hat{\mathbf{t}})\right]\right\|^2.
\end{aligned} \tag{27}$$

[5] Our Matlab code is available at http://www.emt.tugraz.at/~pinz/code/. A C++ implementation is available in ARToolkitPlus [15].

Since $\|\mathcal{R}\mathbf{a}\| = \|\mathbf{a}\|$, the error function of the rotated system is equal to the error function of the orignal system (1).

Appendix B

By writing a decomposition of $RR_z(\phi)$ as a product of three rotations,

$$\begin{aligned} RR_z(\phi) &= R_z(\gamma)R_y(\beta)R_x(\alpha = 0) \\ &= \begin{bmatrix} \cos(\gamma)\cos(\beta) & -\sin(\gamma) & \cos(\gamma)\sin(\beta) \\ \sin(\gamma)\cos(\beta) & \cos(\gamma) & \sin(\gamma)\sin(\beta) \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix}, \end{aligned} \tag{28}$$

we see that the second element in the third row of the composed matrix $(RR_z(\phi))_{32}$ must vanish to obtain a decomposition with $\alpha = 0$. This gives us a constrained and, therefore, a unique solution for the rotation angle ϕ.

$$R_{32}\cos(\phi) - R_{31}\sin(\phi) = 0 \Longrightarrow \phi = \arctan\left(\frac{R_{32}}{R_{31}}\right). \tag{29}$$

Acknowledgments

This work was supported by the Austrian Science Fund (FWF, project S9103-N04 and P15748).

References

[1] B. Wrobel, "Minimum Solutions for Orientation," *Calibration and Orientation of Cameras in Computer Vision,* A. Gruen and T. Huang, eds., chapter 2, Springer-Verlag, 2001.

[2] H. Araújo, R. Carceroni, and C. Brown, "A Fully Projective Formulation to Improve the Accuracy of Lowe's Pose-Estimation Algorithm," *Computer Vision and Image Understanding,* vol. 71, no. 2, pp. 227-238, 1998.

[3] C. Lu, G. Hager, and E. Mjolsness, "Fast and Globally Convergent Pose Estimation from Video Images," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 6, pp. 610-622, June 2000.

[4] M. Fischler and R. Bolles, "The Random Sample Consensus Set: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Comm. ACM,* vol. 24, no. 6, pp. 381-395, 1981.

[5] A. Ansar and K. Daniilidis, "Linear Pose Estimation from Points or Lines," *Proc. European Conf. Computer Vision,* vol. 4, pp. 282-296, May 2002.

[6] P. Wunsch and G. Hirzinger, "Registration of CAD-Models to Images by Iterative Inverse Perspective Matching," *Proc. 13th Int'l Conf. Pattern Recognition,* vol. 1, pp. 78-83, Aug. 1996.

[7] D. Nister, O. Naroditsky, and J. Bergen, "Visual Odometry," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 1, pp. 652-659, 2004.

[8] C. Harris, "Geometry from Visual Motion," *Active Vision,* A. Blake and A. Yuille, eds., chapter 16, MIT Press, 1992.

[9] H. Kato and M. Billinghurst, "Marker Tracking and HMD Calibration for a Video-Based Augmented Reality Conferencing System," *Proc. Second IEEE and ACM Int'l Workshop Augmented Reality,* pp. 85-94, 1999.

[10] S. Malik, G. Roth, and C. McDonald, "Robust 2D Tracking for Real-Time Augmented Reality," *Proc. Conf. Vision Interface,* 2002.

[11] T. Kawano, Y. Ban, and K. Uehara, "A Coded Visual Marker for Video Tracking System Based on Structured Image Analysis," *Proc. Second IEEE and ACM Int'l Symp. Mixed and Augmented Reality,* 2003.

[12] D. Oberkampf, D.F. DeMenthon, and L.S. Davis, "Iterative Pose Estimation Using Coplanar Feature Points," *Computer Vision and Image Understanding,* vol. 63, no. 3, pp. 495-511, May 1996.

[13] J. Garloff and M. Zettler, "Robustness Analysis of Polynomials with Polynomial Parameter Dependency Using Bernstein Expansion," *IEEE Trans. Automatic Control,* vol. 43, no. 3, pp. 425-431, 1998.

[14] R. Moore, *Interval Analysis.* Prentice-Hall, 1966.

[15] ARToolKit Plus, http://studierstube.icg.tu-graz.ac.at/handheld_ar/artoolkitplus.php, 2006.