Using Distributed Sensing and Sensor Fusion for Uncalibrated Visual Manipulator Guidance

Christian Scheering and Bernd Kersting *

**Technical Computer Science, Faculty of Technology, University of Bielefeld, 33501 Bielefeld, Germany*

Abstract. We describe a method for 3D visual manipulator control using a redundant camera system without explicit external or internal calibration. Under the assumption of a simple linear camera model, a fusion equation is derived for which only three parameters have to be estimated, regardless of the number of cameras. Distributed sensor-units provide the necessary measurements, which are fused together in a Kalman filter. In simulations, as well as in real experiments, the feasibility of our approach for a 3D positioning task of a six degree of freedom (DOF) Puma 200 to a target is demonstrated. It is shown that using redundant views increases positioning accuracy and fault tolerance. The achieved accuracy is sufficient to perform an insertion task.

Key Words. uncalibrated vision, distributed sensing, sensor fusion, redundant-camera system

1 Introduction

Using a robot manipulator for an assembly task requires the ability to grasp different parts. The common approach is to solve the 3D relationship between the robot and the environment based upon 2D vision measurements. That in turn requires the internal and external camera parameters to be calibrated which is difficult and cumbersome.

Recently, the idea of uncalibrated visual guidance has attracted more attention. (Skaar *et al.*, 1987) described camera space manipulation while (Yoshimi and Allen, 1996) demonstrated 2D alignment of an eye-on-hand manipulator using rotational epipolar motion. Both (Hager *et al.*, 1995) and (Cipolla and Hollinghurst, 1994) exploit a nearly uncalibrated stereo camera setup. Recent work shows the feasibility using the well known image Jacobian. In (Jägersand *et al.*, 1997) the usefulness of adaptive differential feedback employing a visual motor Jacobian was shown. In (Sutanto *et al.*, 1997) the idea of exploratory movements for dynamic image Jacobian estimation was demonstrated.

This work describes a different way of uncalibrated hand-eye coordination. It is based upon several redundant arbitrary camera views. As stated in (Brooks and

Iyengar, 1996) redundancy increases the sensor reliability, efficiency and performance. The problem is how to fuse the redundant multi-sensor readings properly. We deal with this problem by assigning each camera to a single sensor-unit which provides the specific measurement necessary to guide the manipulator towards a target. Using a parallel (and therefore linear) camera model leads to a simple linear fusion-equation. We show simulations and real experiments demonstrating the capability of a redundant uncalibrated camera system in order to increase position accuracy and in case of different camera failures.

2 Distributed sensing and fusion

The key idea of our visual control is that for a Cartesian motion the image Jacobian is equivalent to the assumption of a parallel-camera model. Defining an image-based position error in j different views and exploiting the parallel projection camera-model leads to a simple linear equation for a resulting Cartesian correction movement – the *fusion equation*. The parameters in turn are estimated with a linear Kalman filter (KF) using measurement obtained by distributed sensors.

2.1 Fusion equation

Many researchers in the field of visual control (either with uncalibrated cameras or not) exploit the so called *image Jacobian* $\boldsymbol{J}$ introduced by (Weiss *et al.*, Oct. 1987) in order to relate a (discrete and small) displace-movement $\Delta \boldsymbol{d}$ (either in joint- or task-space) with a 2 dimensional image-feature displacement $\Delta \boldsymbol{f}$:

$$\Delta \boldsymbol{f} = \boldsymbol{J} \cdot \Delta \boldsymbol{d} \tag{1}$$

The problem is to invert the Jacobian, using a (pseudo) inversion in order to calculate the displacement $\Delta \boldsymbol{d_e}$ corresponding to an image displacement $\Delta \boldsymbol{f_e}$ defined by an appropriate feature-space error function.

We chose a different approach of how to relate a feature-space error function with a corresponding task-space displacement. This approach is somewhat related to the image Jacobian. We use a quite rough approximation of the image-forming process – the parallel projection.

The parallel projection $\boldsymbol{P}^j$ (see (Harris, 1984)) of a 3D world point $\boldsymbol{m}$ in *homogeneous* coordinates $\boldsymbol{m}^w = (m_x, m_y, m_z, 1)^T = (\boldsymbol{m}, 1)^T$ onto the j^{th} camera plane is

$$\begin{aligned} \boldsymbol{f^j} &= \begin{pmatrix} r^j_{11} & r^j_{12} & r^j_{13} & t^j_1 \\ r^j_{21} & r^j_{22} & r^j_{23} & t^j_2 \end{pmatrix} \cdot \boldsymbol{m}^w \\ &= (\boldsymbol{R}^j \ \boldsymbol{t}^j) \cdot \boldsymbol{m}^w \\ &= \boldsymbol{P}^j \cdot \boldsymbol{m}^w \end{aligned} \tag{2}$$

$\boldsymbol{P}^j$ in eq. (2) is simply the first two rows of the corresponding homogeneous transformation ${}^{c^j}\boldsymbol{T}_w$ from the world to the j^{th} camera coordinate system.

The simplest error function for a linear point-to-point movement of a manipulator at $\boldsymbol{m}$ to a goal $\boldsymbol{g}$ is to define an appropriate error-displacement vector $\Delta \boldsymbol{d_e}$ which has to become (nearly) zero.

$$\Delta \boldsymbol{d_e} = \boldsymbol{m} - \boldsymbol{g} \rightarrow 0 \tag{3}$$

For the corresponding displacement feature $\Delta \boldsymbol{f}_e^j$ in the j^{th} camera using eq. (2) follows:

$$\begin{array}{rcl} \Delta \boldsymbol{f}_e^j & = & \boldsymbol{f}_m^j - \boldsymbol{f}_g^j \\ & = & \boldsymbol{P}^j \cdot \boldsymbol{m}^w - \boldsymbol{P}^j \cdot \boldsymbol{g}^w \\ & = & \boldsymbol{R}^j \cdot \boldsymbol{m} + \boldsymbol{t}^j - \boldsymbol{R}^j \cdot \boldsymbol{g} - \boldsymbol{t}^j \\ & = & \boldsymbol{R}^j \cdot \Delta \boldsymbol{d_e} \end{array} \tag{4}$$

Eq. (4) shows additionally that the parallel projection $\boldsymbol{R}^j$ of a displacement is equivalent to the image Jacobian in eq. (1).

Given a set of three Cartesian linear independent displacement vectors[1] $\{\boldsymbol{d}_1, \boldsymbol{d}_2, \boldsymbol{d}_3\}$ the error-displacement vector $\boldsymbol{d_e}$ can be calculated by their linear combination:

$$\boldsymbol{d}_e = \sum_{i=1}^{3} \xi_i \boldsymbol{d}_i \tag{5}$$

Under the assumption of a parallel projection $\boldsymbol{R}^j$ the projected version of eq. (5) is:

$$\begin{array}{rcccl} \boldsymbol{f}_e^j & = & \boldsymbol{R}^j \cdot \boldsymbol{d}_e & = & \boldsymbol{R}^j \cdot \sum_{i=1}^{3} \xi_i \boldsymbol{d}_i \\ & = & \sum_{i=1}^{3} \xi_i \cdot \boldsymbol{R}^j \cdot \boldsymbol{d}_i & = & \sum_{i=1}^{3} \xi_i \boldsymbol{f}_i^j \end{array} \tag{6}$$

Hence we can define the error function as the projection of the corresponding Cartesian displacement $\boldsymbol{d}_e$. Calculating an appropriate set of scalars ξ_1, ξ_2, ξ_3 in the image space and inserting them into eq. (5) leads directly to the desired displacement-vector in the Cartesian 3D space.

Unfortunately eq. (6) is under-determined. Therefore at least two views are necessary yielding an over-determined system. Assuming a redundant multi-camera system with j different sensor-units, all views can be integrated simply by solving the following over-determined system:

$$\underbrace{\begin{pmatrix} \boldsymbol{f}_e^1 \\ \boldsymbol{f}_e^2 \\ \vdots \\ \boldsymbol{f}_e^j \end{pmatrix}}_{\boldsymbol{z}} = \underbrace{\begin{pmatrix} \boldsymbol{f}_1^1 & \boldsymbol{f}_2^1 & \boldsymbol{f}_3^1 \\ \boldsymbol{f}_1^2 & \boldsymbol{f}_2^2 & \boldsymbol{f}_3^2 \\ \vdots & \vdots & \vdots \\ \boldsymbol{f}_1^j & \boldsymbol{f}_2^j & \boldsymbol{f}_3^j \end{pmatrix}}_{\boldsymbol{H}} \cdot \underbrace{\begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{pmatrix}}_{\boldsymbol{\xi}} \tag{7}$$

[1] Because in the following only displacements are considered the Δ is omitted.

Eq. (7) plays the central role in our approach and is called the *fusion equation*. Only three parameters have to be estimated independently of the number of cameras and only three initial test movements are necessary (instead of several exploratory movements when performing a repeated Jacobian acquiring as in (Sutanto et al., 1997)).

2.2 Distributed sensing units

From each camera only the position-residual $\boldsymbol{f}_e^j$ between the goal and the manipulator is necessary in Eq. (7). Therefore we can assign each camera to a sensor-unit which is able to calculate its local $\boldsymbol{f}_e^j$ and send it back on request to a central fusion-unit which in turn solve the parameters of the fusion-equation.

2.3 Solving the fusion equation

We use a linear discrete Kalman filter to solve the parameters of Eq. (7). Assuming zero-mean, white-noise $\mathbf{v}$ and $\mathbf{w}$, the plant and measurement equation are:

$$\begin{array}{rcll} \xi(k+1) & = & \xi(k) + \mathbf{v}, & \mathbf{v} \sim N(0, \mathbf{Q}) \\ \mathbf{z}(k) & = & H(k) \cdot \xi(k) + \mathbf{w}, & \mathbf{w} \sim N(0, \mathbf{R}) \end{array} \tag{8}$$

The incremental prediction and update solutions can be found in (Bar-Shalom and Li, 1993). In our approach the whole system dynamic is included in the system noise $\mathbf{v}$. We have chosen pure diagonal matrices for $\mathbf{Q}$, $\mathbf{R}$ and the initial state covariance $\mathbf{P}_{(0|0)}$ with the diagonal elements $\sigma^2_{P_{(0|0)}} = 0.1, \sigma^2_Q = 0.01$ and $\sigma^2_R = 5.0$. The initial state-estimate is set to $\boldsymbol{\xi}_{(0|0)} = (1,1,1)^T$.

For a point-to-point movement to a selected target the manipulator first makes three Cartesian test movements. Each sensor-unit detects their image $\boldsymbol{d}_i^j$ and send them back to the fusion-unit. With their corresponding position-residuals an initial down-scaled correction movement $\boldsymbol{d}_c = s \cdot \boldsymbol{d}_e, 0 < s < 1$ is calculated. After each movement a new $\boldsymbol{\xi}$ is estimated by asking each sensor-unit for the actual position-residual. This is iterated as long as the target is not reached.

3 Simulations

In the simulations the system-behaviour using redundant cameras and its robustness in potential failure situations is investigated. The task is to position the manipulator tool center point at a target position. The images of these points are generated using a pin hole camera model for each view. However, for the algorithm the projected points are used only and not the information of the simulated camera. This is still an idealisation since in reality there is no guarantee that the measured points in the images are the projection of the same 3D point. At least they should be closed neighbours.

Each measurement of the target- and manipulator-position is overlayed with 2 dimensional Gaussian noise with a variance of 5 in both horizontal and vertical

direction. In the simulation setup each test move has a 50mm length aligned with the robots coordinate system. The distance to be moved is about 375mm. Each camera has a distance of approximately 2m from the scene. The used pin-hole cameras have a uniform scaling of 70 pixel/mm and a focal length of 20mm.

In order to show that even under the assumption of a parallel projection our iterative approach still holds, the parallel camera model is not used to simulate the feature generation. At present three different termination criteria have been used:

- Maximum number of iterations $I_{max} = 200$.
- Two dimensional minimal distance d_2; the approach is stopped if in every image the projection of the moved distance is less than d_2 pixel.
- Three dimensional minimal distance d_3; the approach is stopped if the last three real motions have been less than d_3 mm each.

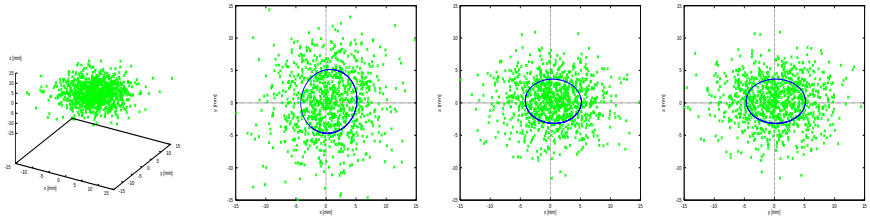
The experiments were run 1000 times each with two cameras one observing the xz-plane and the other one the yz-plane. Tab. 1 shows the results for different termination criteria. The number of runs n_r with successful termination due to the criterion, the corresponding mean number of iterations n_i and the mean 3D residual distance d after termination are displayed. For those runs which were terminated by exceeding the iteration limit the max-iteration residual d_m is shown, too.

		Criterion				
		d_2		d_3		d_2 or d_3
		2_p	5_p	3_{mm}	5_{mm}	5_p or 5_{mm}
KF	n_r	380	1000	573	1000	1000
	n_i	97	32	92	28	22
	d/d_m	6.4/7.5	6.9	6.0/8.3	6.6	6.9

Table 1 Comparing different termination criteria.

For both termination criteria d_2 and d_3 the (trivial) observation is that the weaker the criterion, the more it fires. However, weakening the criteria does not increase the mean target-distance d significantly. In order to have a criterion that (nearly) always fires and yields a (nearly) minimal number of iterations and a (nearly) minimal residual distance we suggest a combined criterion, shown in the last column of Tab. 1. Although it does not produce the best residual distance it provides the minimal number of iterations.

An example of the end-positions distribution using a Kalman filter with the combined criterion 5p or 5mm is shown in Fig. 1. The target position has been transformed into the origin. Each ellipse is equivalent to the standard deviation calculated from the covariance of the appropriate distribution. The ellipse is centered at the mean value of the distribution and is oriented along the principal axis of the covariance of the distribution. It can be seen in Fig. 1(c) and (d) that the distribution around the z-axis is more compact than the distribution around the x- and y-axis.



(a) 3D Points (b) yx-plane (c) zx-plane (d) zy-plane

Figure 1 End position distribution using 2 cameras.

This is due to the fact that in this simulation the z-axis had been observed by both cameras.

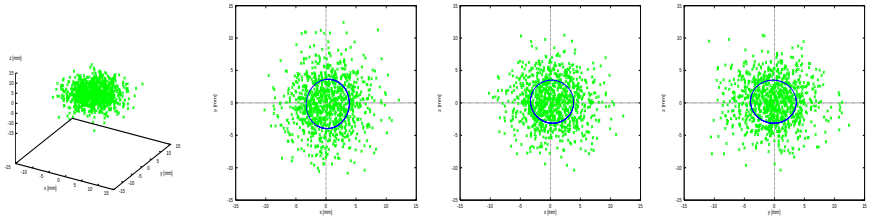
Therefore we should expect better results (i.e. less iterations, less d and more compact distributions) if a redundant third camera is introduced observing the xy-plane. This is shown in Fig. 2 using the same combined termination criterion. The denser distribution is obvious – the deviation ellipses are nearly circles and have become smaller. Comparing the results for n_i and d for 1000 runs (as shown in Tab. 2) it can be seen that both the mean residual distance and the mean number of iterations decreases for d_3 and the combined criterion.

		Criterion				
		d_2		d_3		d_2 or d_3
		2_p	5_p	3_{mm}	5_{mm}	5_p or 5_{mm}
n_i	2 Cameras	161	32	138	28	22
	3 Cameras	196	84	78	19	19
	Diff[%]	+22	+163	-43	-32	-14
d[mm]	2 Cameras	7.1	6.9	7.0	6.6	6.9
	3 Cameras	6.0	5.4	5.0	5.6	5.7
	Diff[%]	-15	-22	-29	-15	-17

Table 2 Comparing n_i and d for a KF solution between 2 and 3 Cameras.

3.1 Defect simulation

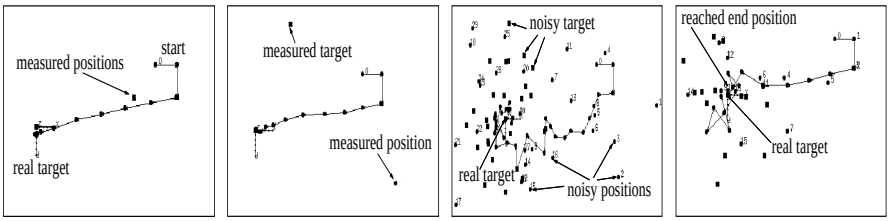
Three different failure types of a single camera in a set of three have been simulated. The first is that both the target and the manipulator have always the same position. In this situation no residual information from this camera is obtained but the target is reached (see Fig. 3(a)). The second failure is that target and manipulator have always the same but different positions. The residual is always the same and non-zero but the target is reached, too (see Fig. 3(b)). In the last case both the target and manipulator position are very noisy. The problem is that the (very important)



(a) 3D Points (b) yx-plane (c) zx-plane (d) zy-plane

Figure 2 End position distribution using 3 cameras.

test movements are detected with heavy noise, too. The worse they are detected, the worse the positioning is (see Fig. 3(c)). If the test movements are detected without or with less noise (e.g. by a repetition of every move and calculating the mean) the result is improved (the target is reached after 16 iterations, see Fig. 3(d)). In order to increase the robustness of the system in the case of a camera defect, these results suggest the use of redundant cameras which are fairly easy to incorporate in our approach.



(a) (b) (c) (d)

Figure 3 Different failure situations.

4 Experiment

In this section we demonstrate the quality of our approach in a real experiment. The manipulator is a 6 DOF Puma 200 using RCCL (Lloyd, 1988) as the control language.

The cameras are in approximately 1.5m distance. The target is a hole with radius 8mm in a wooden toy cube. The manipulator carries another cube with a peg which has to be inserted. The center of the hole is at $(-30, 340, 163)$mm and the manipu-

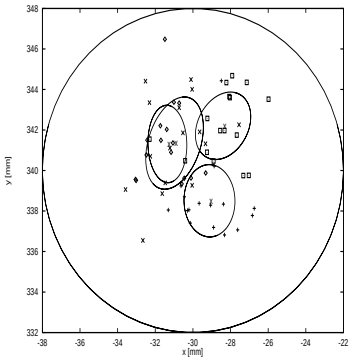


Figure 4 End points in reality.

series	d	n_i
1	2.5	12
2	2.5	11
3	2.5	9
4	3.0	9
mean	2.6	10

Table 3 Mean real end position residual and iterations.

lator is at $(150, 250, 0)$mm. Four test series containing 64 runs have been performed using a combined termination criterion d_2 = 2p and d_3 = 1mm.

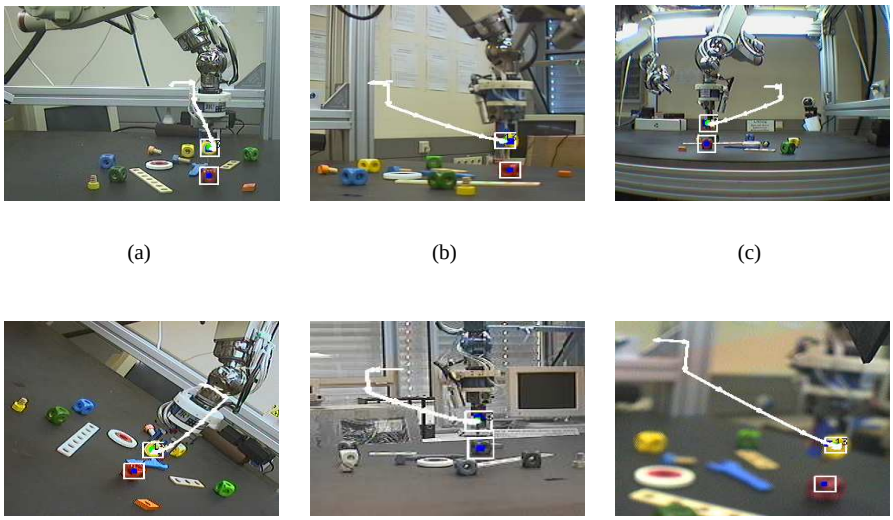
Before running a test the fusion-unit requests an actual image from each sensor-unit. The target and the manipulator point to be tracked are marked by the user resulting in a template for both in each image. The tracking itself is performed by each sensor-unit exploiting simple template matching. At each new control-cycle the fusion-unit requests all the appropriate position-residuals from each sensor-unit. Due to the selection procedure and the different perspectives of each camera the template centers are not the projection of the same point in 3D space.

For security reasons a point above the selected target describes the desired target position. For our setup this relative correction vector is $\Delta \boldsymbol{c} = (0, 0, -50)$mm. This relative distance is projected onto each image using the parallel-camera model in eq. (4). The six parameters are calculated based on the measured projection of the test movements. Despite these errors (noise, parallel projection, manual target selection) the results shown in Tab. 3 for the mean target residual distance d and the mean iteration number n_i are satisfying. The hole was found in all runs and the mean distance is approximately 2.6mm from the center of the hole. Fig. 4 shows the corresponding distribution of end positions of all 64 runs. Each ellipse around the mean value is equivalent to the distance standard deviation of a test series. With the achieved accuracy the peg was inserted successfully simply by moving downward with a force-guarded motion.

The last series of images in Fig. 5 demonstrates the ability of our approach to fuse several arbitrary positioned camera views even if some images have poor quality due to high lens distortion (Fig. 5(c)) or blur (Fig. 5(f)).

5 Conclusions

This work presented an uncalibrated visual manipulator control using redundant cameras. A parallel-camera model is used to calculate a correction. Instead of



(a) (b) (c)

(d) (e) (f)

Figure 5 End positions and projected trajectory in six arbitrary views.

exploiting the Jacobian directly, a linear combination of three linearly independent test movements is performed. Independently of the number of cameras only three parameters have to be estimated. Using a redundant camera-system and exploiting distributed sensing robustness and performance are increased. The results of the distributed sensing-process are fused together with a Kalman filter. The quality of this approach is shown in simulations and real experiments.

The next step in this framework is to incorporate an automatic motion detection and tracking ability. Another point is to apply the known robot motion in order to estimate the pin-hole camera parameters without any further knowledge. Using this model, an estimate of the epipolar geometry might be useful in order to detect a target which has been selected in only one view. Orientation control will be examined using additional track points on both the target and manipulator.

More work will go into flexibilisation – instead of using a fixed set of a priori known sensor-units a dynamically self-configuring sensor-unit network could be possible, using for instance a Contract Net Protocol (see (Smith, 1981)). Also the autonomy of each sensor-unit could be increased exploiting the idea of a decentralised Kalman filter as in (Brown et al., 1992).

Acknowledgement

The work described in this paper has been funded by the German Research Foundation (DFG) in the project SFB 360.

6 REFERENCES

Bar-Shalom, Y. and X. Li (1993). Estimation and Tracking. Artech House.

Brooks, R. R. and S. S. Iyengar (1996). Maximizing multi-sensor system dependability. In: IEEE/SICE/RSJ Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems. pp. 1–8.

Brown, C., H. Durrant-Whyte, J. Leonard, B. Rao and B. Steer (1992). Distributed data fusion using kalman filtering: A robotics aplication. In: Data Fusion in Robotics and Machine Intelligence (M. A. Abidi and R. C. Gonzales, Eds.). pp. 267–309. Academic Press.

Cipolla, R. and N. Hollinghurst (1994). Uncalibrated stereo hand-eye coordination. Image and vision computing **12**(3), 187–.

Hager, G., W. Chang and A.S. Morse (1995). Robot feedback control based on stereo vision: Towards calibration-free hand-eye coordination. IEEE Control Systems Magazine **15**(1), 30–39.

Harris, D. (1984). Computer graphics and applications. Chapman and Hall.

Jägersand, M., O. Fuentes and R. Nelson (1997). Experimental evaluation of uncalibrated visual servoing for precision manipulation. In: Proc. IEEE Int. Conf. Robot. Automat.. pp. 2874–.

Lloyd, J. (1988). RCCL User's Guide. Computer Vision and Robotics Laboratory.

Skaar, S., W. Brockman and W. Jang (1987). Camera-space manipulation. Int. Jour. Robot. Research **6**(4), 20–32.

Smith, Reid G. (1981). Distributed Problem Solving. UMI Research Press.

Sutanto, H., R. Sharma and V. Varma (1997). Image based autodocking without calibration. In: Proc. IEEE Int. Conf. Robot. Automat.. pp. 974–.

Weiss, L. E., A. C. Sanderson and C. P. Neumann (Oct. 1987). Dynamic sensor-based control of robots with visual feedback. IEEE Trans. Robot. Automat. **RA-3**, 404–417.

Yoshimi, B. and P. Allen (1996). Alignment using an uncalibrated camera system. IEEE Trans. Robot. Automat. **12**(5), 516–521.