Robotica (2005) volume 23, pp. 409–418. © 2005 Cambridge University Press
doi:10.1017/S0263574704000840 Printed in the United Kingdom

A CAD model based tracking system for visually guided microassembly

Kemal Berk Yesin and Bradley J. Nelson

Institute of Robotics and Intelligent Systems, Swiss Federal Institute of Technology (ETH), Zurich (Switzerland)

SUMMARY
We present a visual feedback method for closed loop control of automated microassembly. A CAD model based multi-camera visual tracking system that is well suited for flexible automation and assembly of complex 3D geometries was developed. The system is capable of providing six degree-of-freedom pose feedback on the observed micro-components in real-time (30 Hz). Using CAD models of the observed objects, a complete description of the observed scene, including the effect of occlusions, is available and dependence on distinctive visual features such as fudicial marks is avoided.

KEYWORDS: Visual tracking; Visual servoing; CAD; Microassembly; Micromanipulation.

1. INTRODUCTION

1.1. Motivation

There is a growing interest in MEMS devices that are built by assembling individual microcomponents. A strong motive for microassembly is the demand for hybrid MEMS that can combine incompatible materials or manufacturing processes (e.g. CMOS, MOEMS). Microassembly can also overcome the planar, 2.5D shape limitations of standard MEMS manufacturing processes and create complex 3D geometries. In addition, microassembly can increase the overall yield as the number of layers and manufacturing steps grow for a complex device. For example, the electromagnetic MEMS linear actuator described in reference [1] uses an externally wound 400 turn microcoil with an electroplated structure to achieve high force and displacement capability. Traditional coilwinding methods provide greater yield than a potential multilayer MEMS process would for producing such high winding density microcoils due to the defects that arise in microfabricating the coil.

The majority of robotic assembly operations in the macro-world rely on accurate robots that play back recorded motions. However, this form of open-loop manipulation is not suitable at the microscale due to the increased precision requirements and the vastly different mechanics of manipulation. While gravity is the dominant force in the macro domain, electrostatic forces, surface tension effects due to humidity and intermolecular Van der Waals forces become dominant at the microscale, acting as external disturbances to the microassembly process.[2] Closed-loop control through computer vision feedback has been applied to overcome some of these problems and various methods and systems for visually guided microassembly have been presented in the literature.

In this paper, we present an alternative method for visual tracking of MEMS micro-components using a CAD model based multi-camera vision approach. Our goal is to develop a generally applicable feedback method that is better suited for the flexible automation of microassembly processes. Section 1.2 reviews the previous research on the topic. Section 2.1 discusses the advantages of the model based approach for micro-scale vision. Section 2.2 presents the algorithm in detail and Section 3 presents the experimental results.

1.2. Previous research

The guidance of robots through real-time and continuous visual feedback is generally known as *visual servoing*, and the continuous observation of the objects of interest is referred to as *visual tracking*. Visual tracking of an object involves the detection of some known object *features* in the acquired images and the estimation of the object position and orientation using these features. A feature can be a distinctive part of the object and can exist naturally as part of the geometry (i.e. a corner, an edge), or as a deliberately fabricated marking on the surface (i.e. a fudicial mark).[3]

Various systems have been presented in the literature for visually guided microassembly;[4–14] however, few are intended for industrial level, flexible automation of microassembly.[15] In the majority of these, features are user-selected from images and stored as templates during an initialization stage, although methods for automatic selection of features have also been suggested.[6,16–18] In reference [4], a detailed model of the microscope optics was used together with CAD models of the objects to create synthetic images which were then used for automatic selection of features. During the tracking stage, the locations of features are typically found by matching the previously stored feature templates to the current images using correlation based techniques, such as SSD.[19]

As will be discussed in more detail in Section 2.1, microscope optics approximate an orthographic projection system and do not reflect changes in the image scale as objects move along the view axis. Therefore, only three degrees of

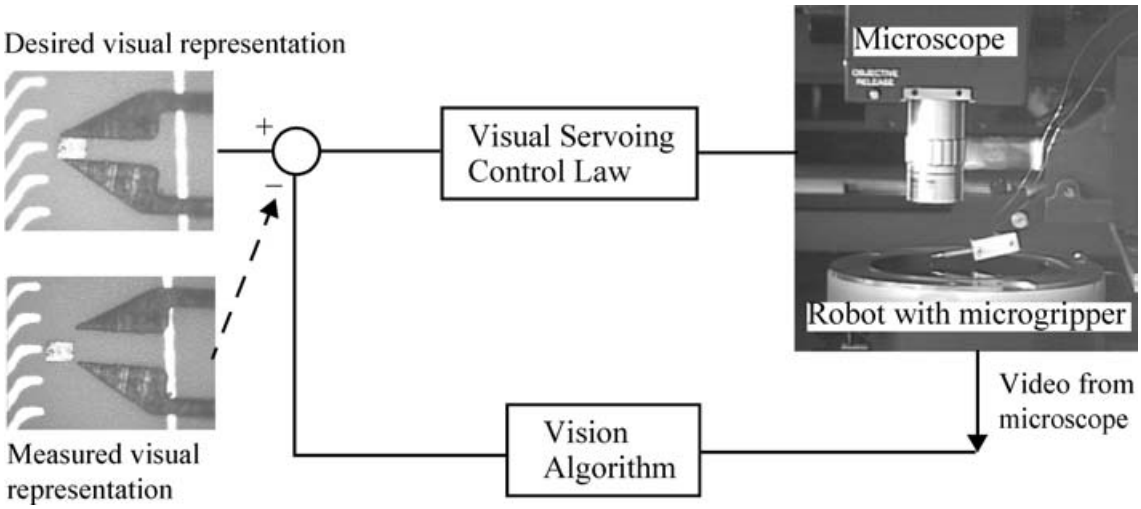

Fig. 1. Closed-loop control of microassembly through visual feedback.

freedom (translations parallel to the image plane and rotation about the optical axis) of object motion are observable. A common approach in microscopic vision to resolve depth along the view axis is the depth from defocus method.[6,8,20] This method uses a visual metric of image sharpness to detect, whether points or areas of interest are in focus, as the focus point of the lens is moved along the optical axis in controlled increments. This way the relative distance of two points at different depths can be found by determining where along this "focal-scan" the points are in best focus. However, the focal-scan method is time-consuming and prevents real-time application. A structured illumination method suitable for planar components has also been investigated as an alternative to depth from defocus.[7]

Virtually, all examples in the literature have focused on 4-DOF (degree-of-freedom) tracking where a single, top-view microscope observes the position and orientation of the components on a plane (Figure 1) and the depth from defocus technique is used to resolve depth. However, even with planar manipulation tasks, the effects of microscale forces were shown to create out-of-plane rotation errors.[8] For this reason, 6-DOF manipulation and feedback requirements can be expected in order to build complex, 3D hybrid MEMS devices.

2. CAD MODEL BASED MULTI-CAMERA VISUAL TRACKING WITH MICROSCOPES

2.1. Advantages of the CAD model based approach

The visual feedback system that is presented here was developed as an essential component of a flexible automation system for microassembly. A CAD model based multi-camera tracking approach was taken to overcome the difficulties associated with similar systems in the literature, as reviewed above. There are several advantages to using CAD models as a standard form of input for a flexible automation system and a CAD model based vision has been applied to macro-scale robotics.[21–24] In general, the CAD model of a MEMS component is readily available from its design phase. More importantly, CAD models provide 3D information on component geometry so that appearance can be predicted for any viewpoint while considering the effect of occlusions. This is essential for a visual tracking system that can handle translations and rotations about an arbitrary axis. In addition to these advantages, by using object edges (contours) as visual features to be observed, the CAD model based approach avoids the need for fudicial marks or other distinctive features to be present on the object. CAD models of the components and the assembly have also been used for automated path and assembly planning for macro-scale robotics.[25] A CAD model representation is a natural choice of input for a similar automated planning system for microassembly, which could also include the effects of microscale forces in planning.

An important but not obvious advantage of the model based multi-camera approach, especially in comparison to stereo vision, becomes more significant when working with microscopes. This advantage is the possibility of positioning multiple microscopes around the workspace without constraints on their relative locations. The concept of *visual resolvability*[26] indicates the effect of camera placement in a multi-camera system. A camera projects the 3D features P_i of an object onto the image plane and presents them as 2D image features p_i, which can be modelled by perspective projection

$$P_i = [X_i\ Y_i\ Z_i]^T \quad p_i = [u_i\ v_i]^T = \left[\frac{fX_i}{Z_i}\ \frac{fY_i}{Z_i}\right]^T \qquad (1)$$

where f is the focal length of the camera lens system. Although the depth information is lost for a single 3D feature point, object pose can be determined using three or more features. However, not all degrees of freedom of the object can be resolved equally well. In other words, the vision system is not equally sensitive to motion along any direction. With n features of an object being observed by all cameras, the $2n \times 6$ image Jacobian matrix J, which defines the mapping between a given 6 DOF differential motion vector ∂h of an object and the resulting differential motion ∂p of 2D features on the image planes of the cameras, represents

https://doi.org/10.1017/S0263574704000840 Published online by Cambridge University Press

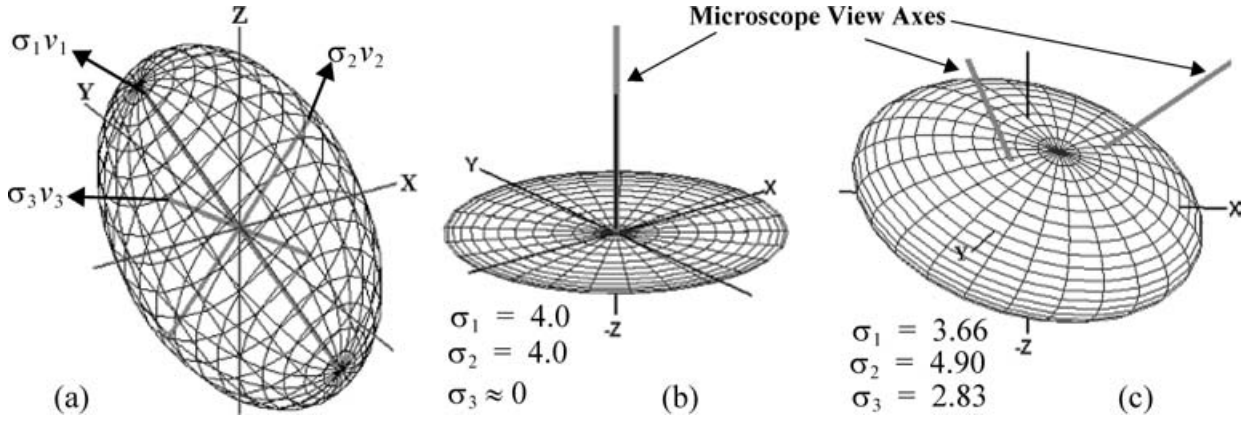

Fig. 2. (a) Visual resolvability ellipsoids show the relative sensitivity of a system along different directions in the task space. Ellipsoids for translations with (b) one microscope and (c) two microscopes.

this anisotropic sensitivity.

$$J_i = \frac{\partial p_i}{\partial h} = \begin{bmatrix} \frac{f}{Z_i} & 0 & -\frac{fX_i}{Z_i^2} & -\frac{fX_iY_i}{Z_i^2} & f+\frac{fX_i^2}{Z_i^2} & -\frac{fY_i}{Z_i^2} \\ 0 & \frac{f}{Z_i} & -\frac{fY_i}{Z_i^2} & f+\frac{fY_i^2}{Z_i^2} & -\frac{fY_iX_i}{Z_i^2} & -\frac{fX_i}{Z_i} \end{bmatrix}_{(2\times 6)}, \tag{2}$$

$$J = \begin{bmatrix} J_i \\ \cdots \\ J_n \end{bmatrix}_{2n\times 6}, \quad \partial p = \begin{bmatrix} \partial p_i \\ \cdots \\ \partial p_n \end{bmatrix}_{2n\times 1}, \quad \partial p = J\,\partial h$$

By separating the Jacobian into translational and rotational parts as $J = [J_v\ J_\omega]$, two 3D *visual resolvability ellipsoids* can be generated to picture this direction-dependent sensitivity. Let $J_v = U\Sigma V^T$ be the singular value decomposition (SVD) of the $2n \times 3$ Jacobian for translations. $U_{2n\times 2n}$ and $V_{3\times 3}$ contain the unit length eigenvectors of $J_v J_v^T$ and $J_v^T J_v$, respectively, and Σ is a diagonal matrix containing the squareroots of the eigenvalues of $J_v J_v^T$ and $J_v^T J_v$, also called the singular values of J_v, in descending order such that $\sigma_1 > \sigma_2 > \sigma_3 \geq 0$. Notice that by this representation a given 3D velocity vector of the object is decomposed into three orthogonal components along the columns v_1, v_2, v_3, of V, and each component is scaled by the singular values $\sigma_1, \sigma_2, \sigma_3$, respectively. Therefore, the largest image-space motion is generated by motion of the object along v_1, and similarly, v_3 is the direction along which the vision system is least sensitive. A visual resolvability ellipsoid (Figure 2a) with primary axes along $\sigma_1 v_1$, $\sigma_2 v_2$, and $\sigma_3 v_3$ represents this direction dependent sensitivity. A similar ellipsoid can be generated for rotations of the object using J_ω.

Figures 2b and 2c show the resolvability ellipsoids for translations for one microscope and two identical microscopes with 2X magnification observing a square shaped object with 500 μm sides. In the single microscope case the length of axis of the ellipsoid along the view axis approaches zero indicating insensitivity along that direction due to orthographic projection properties of microscope lenses. Unlike perspective projection, the orthographic projection of a feature point is not sensitive to changes in depth, and, therefore, motion along the view axes is not observable. Numerically, this can be simulated by using a large focal length and camera distance compared to the size of the object that is observed. With two cameras, however, the sensitivity is much more uniform and motion along any direction can be observed. In effect, each microscope compensates for the other's insensitivity along the view axis. However, the microscope view-axes have to be largely separated to achieve this quality. This is why the model-based approach is advantageous over stereo vision since it does not restrict the relative positioning of the cameras. In stereo vision (Figure 3a), a correspondence match is made for the same feature between different images and the feature's 3D location is solved by triangulation. Therefore, it is necessary to have the same features visible in both views, restricting how much the cameras can be separated. The model based method (Figure 3b), on the other hand, makes matches between 3D model points and 2D image features independently for each camera image. For this reason, it is possible to separate the viewpoints of the cameras by a relatively large amount in order to maximize resolvability.

2.2. The tracking algorithm

Figure 4 illustrates the steps of the CAD model based visual tracking method performed on a single object. The same steps are repeated for each modeled object. Note that all the computationally expensive model and image processing steps are done independently for each camera, until the final pose estimation step where the point data from multiple cameras are combined. Since the large image data is not shared, this method is suitable for parallel implementation and the number of cameras simultaneously observing the scene can be easily increased to improve the robustness and precision of the overall system.

The CAD models that are input to the system are boundary representation (BREP) models drawn in a commercial CAD package. The BREP model consists of a mesh of triangular patches that cover the outer surface of the solid object. At the vision preprocessing step, performed off-line, the model is translated into an internal format that also classifies the triangle edges as "invisible", "silhouette" or "sharp" according to the surface curvature of the object at the edge location. Invisible edges are frequently encountered when a triangular mesh representation of a flat surface is generated whereas sharp edges typically occur between the top and side

https://doi.org/10.1017/S0263574704000840 Published online by Cambridge University Press

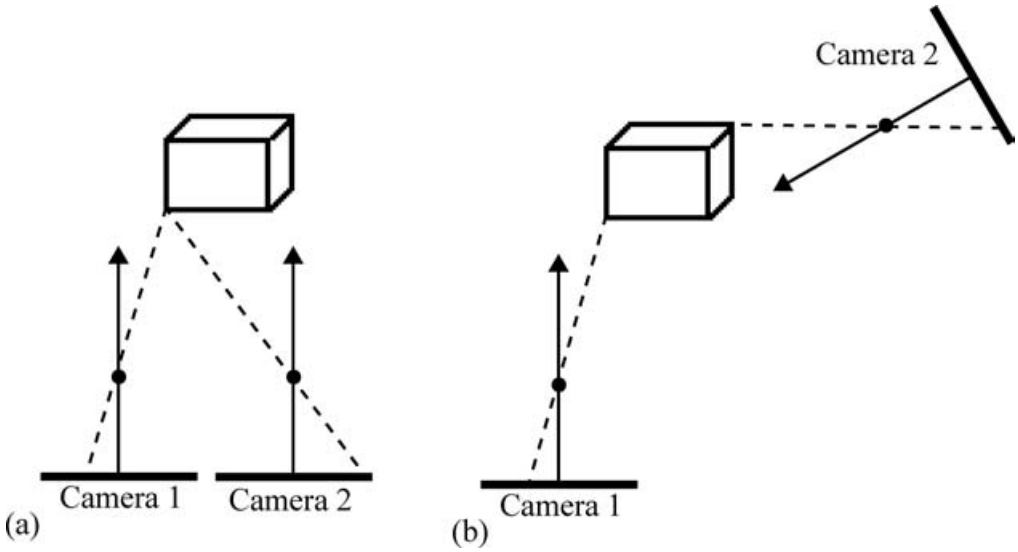

Fig. 3. (a) Stereo (b) model based multi-camera vision.

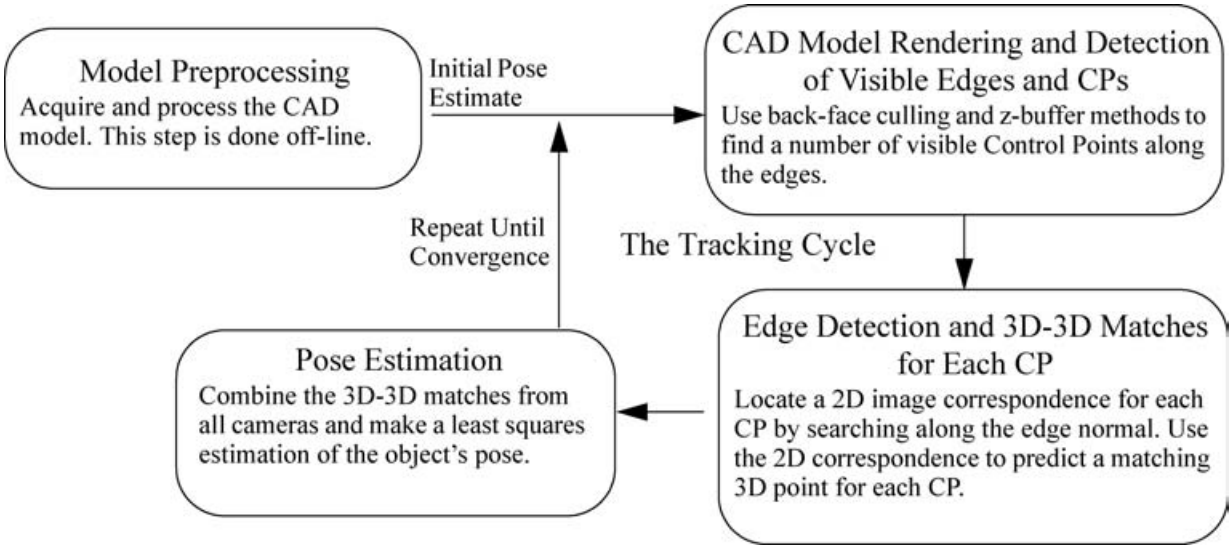

Fig. 4. Steps of the CAD model based tracking method.

faces of planar MEMS components. Notice that the triangular mesh representation and the edges are an approximation of surfaces/contours with smooth curvatures, and the level of approximation is adjustable by the user. Therefore, the model based method is not specific to polyhedral objects, and round objects like glass fibers can also be tracked.

The tracking cycle begins with the rendering of the CAD model using the previously calibrated camera parameters and the initial estimates of their poses. These estimates can be the result of an initial object recognition/localization step if the tracking cycle is just starting, or they can be the results from the previous cycle. The next step is the determination of visible edge segments that will be used for tracking. For each object, the list of edges is processed and the edges that are within the field of view and are potentially visible (i.e. a sharp edge or a silhouette edge that is actually part of the object silhouette) are determined. Some of these edges that are on the back side of the object and, therefore, are self-occluded are eliminated using the backface culling method.[27] A final visible edge detection is performed using the z-buffer (depth buffer) algorithm[27] taking advantage of specialized hardware support for this method offered by most commercial graphics cards. The z-buffer algorithm uses a depth buffer in addition to the color buffer for the image to store a depth value (i.e. distance to the image plane along the camera view axis) associated with every pixel. The visibility of a 3D model point can be determined by comparing its depth value to the corresponding value in the buffer. Those edges that were not eliminated by back-face culling are divided into small (e.g. ~5 pixel long) edge segments that are represented by a *control point* (CP_i) at the middle of the segment (Figure 5a). While CP_i^O, described in the object local coordinate frame, is the 3D control point, CP_i^I is its 2D image space counterpart found by projecting the CP_i^O according to the last known pose of the object. The visibility of the control point CP_i is assumed to indicate the visibility of the ith edge segment.

https://doi.org/10.1017/S0263574704000840 Published online by Cambridge University Press

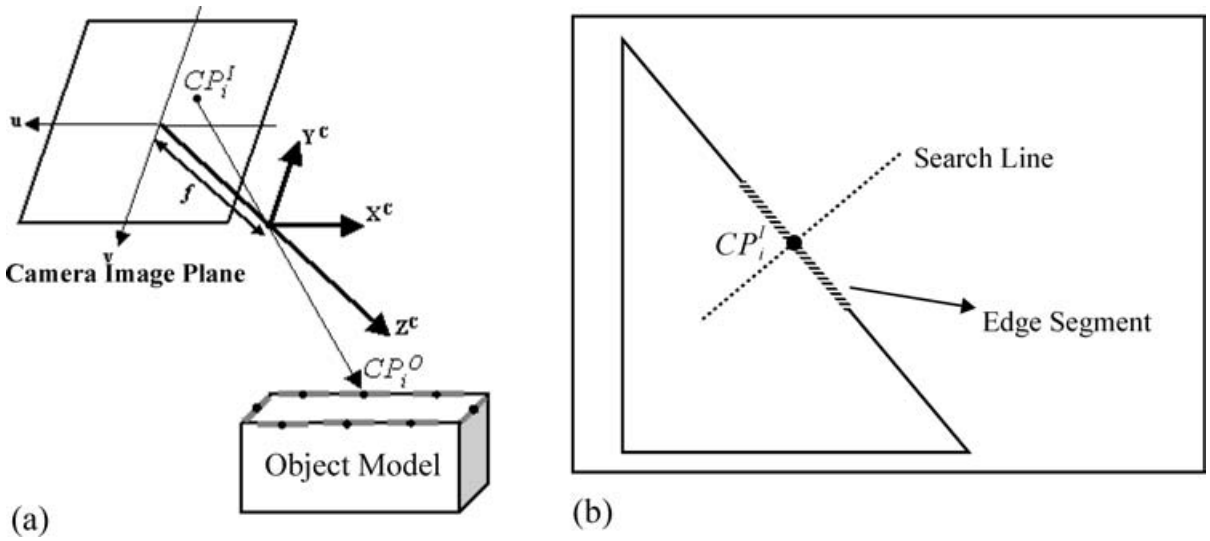

Fig. 5. (a) Visible edge segments are represented by control points (b) Search line for a control point along the normal to the edge in the image.

The next step in the tracking cycle is local edge detection to find matches between visible edge segments and their correspondences in the video image. The edge segment is represented in the image by CP_i^I. Since the location of CP_i^I in the current video frame is assumed to be within a close neighborhood of its last known location (i.e. its location in the previous video frame), the edge detection and matching step can be narrowed to a small search window around the edge segment. The search window is further reduced to a line by taking advantage of the aperture problem which states that the component of motion of an edge that is tangential to itself is not locally observable (i.e. observable at a point on the edge). Only the component along the normal to the edge can be detected. Therefore, the search for the matching edge segment is done along a 1D search path normal to the segment. Figure 5b illustrates an edge segment and the search line along the normal. In actual implementation, a quasi-normal direction chosen among four principle directions at 45° intervals is used instead of the actual normal for computational efficiency.

The edge detection process is similar to the one used in reference [23]. The search line consists of m pixels along the quasi-normal $\vec{n}$, centered at the CP_i^I (3),(4) (Figure 5b). At each pixel e_k along the search line (4), the normal and tangential components of the image gradient are found using the 1-dimensional $[-1\,0\,1]$ kernel. A gradient index G_k is computed by subtracting the absolute value of the normal component from the absolute value of the tangential component (5). If the gradient index value is higher than a positive threshold, that pixel's location on the searchline is added to a list L_i of possible matches for CP_i^I (6). After all pixels of the search line are processed, the mean $\boldsymbol{M}(L_i)$ (7) and standard deviation $\sigma(L_i)$ (8) of the elements of L_i are found. The mean is assigned as the matching image point $CP_i'^I$ for the model control point CP_i^I (9).

$$CP_i^I = [u\ v]^T \tag{3}$$

$$e_k = CP_i^I + k\vec{n}, \quad k = -(m/2)\ldots(m/2) \tag{4}$$

$$G_k = \left|\frac{\partial}{\partial \vec{n}} I(e_k)\right| - \left|\frac{\partial}{\partial \vec{t}} I(e_k)\right| \tag{5}$$

$$G_k > Threshold \Rightarrow k \in L_i \tag{6}$$

$$\boldsymbol{M}(L_i) = \frac{\sum_{s=1}^{size(L_i)} L_i(s)}{size(L_i)} \tag{7}$$

$$\sigma(L_i) = \frac{\sum_{s=1}^{size(L_i)} |\boldsymbol{M}(L_i) - L_i(s)|}{size(L_i)} \tag{8}$$

$$CP_i'^I = CP_i^I + \boldsymbol{M}(L_i)\vec{n} \tag{9}$$

$$w_i = \frac{1}{\sigma(L_i)} \tag{10}$$

The scalar weight w_i, calculated from the standard deviation (10), is later used as a measure of confidence at the pose estimation step. A high standard deviation indicates that multiple locations on the search line not clustered together had gradient values above the threshold, possibly indicating another nearby and parallel edge. Figure 6 shows a MEMS

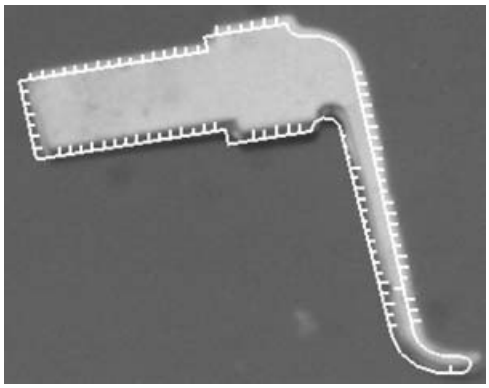

Fig. 6. Image space errors along the edge normals.

https://doi.org/10.1017/S0263574704000840 Published online by Cambridge University Press

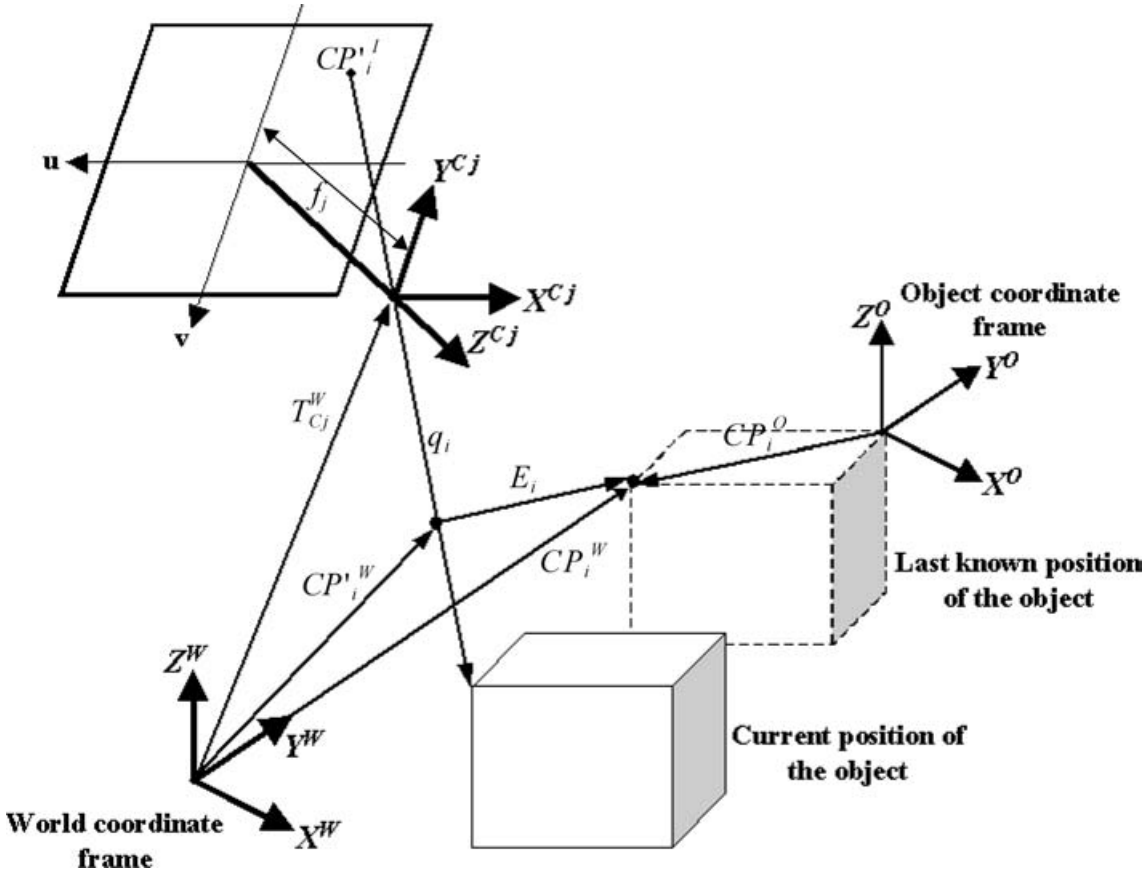

Fig. 7. Object space error definition for multi-camera tracking.

component with its CAD model superimposed. The short lines indicate the image space errors between the model points and the corresponding edge points detected in the image.

The last step of the tracking cycle is the pose estimation which is performed after all correspondences between 3D model points CP_i^O and their matching 2D image points $CP_i'^I$ are established. Various methods for pose estimation from 3D-2D correspondences have been reported.[28] The method used here transforms the image space matches $CP_i'^I$ to appropriate object space data points and performs a 3D-3D fitting of the corresponding model and data points in object space.[22,29] Figure 7 shows the geometry of the problem. Let rotation matrix R_{Cj}^W and translation vector T_{Cj}^W describe the pose of the jth camera with respect to the world coordinate frame, while f_j is the corresponding focal length. Similarly, the last known pose of the object is described by R_O^W and T_O^W. For a given 3D model point CP_i^O that is observed by the jth camera, the matching 3D point $CP_i'^W$ is found by projecting the model point onto the line-of-sight vector q_i defined by $CP_i'^I$. The orthogonal projection $Q_i \in \Re^{n\times n}$ that projects a vector onto the space spanned by $q_i \in \Re^n$ is given by

$$Q_i = \frac{q_i q_i^T}{q_i^T q_i} \tag{11}$$

The line of sight vector, expressed in the world coordinate frame is

$$CP_i'^I = [u'\ v']^T \quad q_i = R_{Cj}^W [u'\ v'\ f_j]^T \tag{12}$$

The matching 3D point and the object space error term E_i are given by

$$CP_i^W = R_O^W CP_i^O + T_O^W \quad CP_i'^W = Q_i\left(CP_i^W - T_{Cj}^W\right) + T_{Cj}^W$$
$$E_i = CP_i^W - CP_i'^W = (I_3 - Q_i)\left(CP_i^W - T_{Cj}^W\right) \tag{13}$$

The remaining problem is to find a rotation and translation of the CAD model that minimizes the errors in a least squares manner. Various closed form solutions to this problem have been proposed[30] that use quarternion representation of rotations,[31,32] polar decomposition[33] or the singular value decomposition method[34] to compute the rotation matrix R_O^{*W} directly. An experimental comparison of these methods[35] has shown that the SVD method provides the best overall stability against degenerate point data sets (i.e. as all points approach to a plane, line or point) and the best accuracy and is the method used here.

In the SVD method, a correlation matrix K is calculated as

$$K = \sum_{i=1}^{n} w_i \left(CP_i'^W - \overline{CP'}^W\right)^T \left(CP_i^W - \overline{CP}^W\right) \tag{14}$$

where w_i is the weighting term (10) that was found during the edge detection step, and, $\overline{CP'}^W$ and $\overline{CP}^W$ are the centroids of the n pairs of 3D points collected from all cameras. Let $K = U\Sigma V^T$ be the SVD of K. The best fitting rotation matrix R_O^{*W} and translation vector T_O^{*W} are then found by

$$R_O^{*W} = UV^T \quad T_O^{*W} = \overline{CP'}^W - R_O^{*W}\overline{CP}^W \tag{15}$$

https://doi.org/10.1017/S0263574704000840 Published online by Cambridge University Press

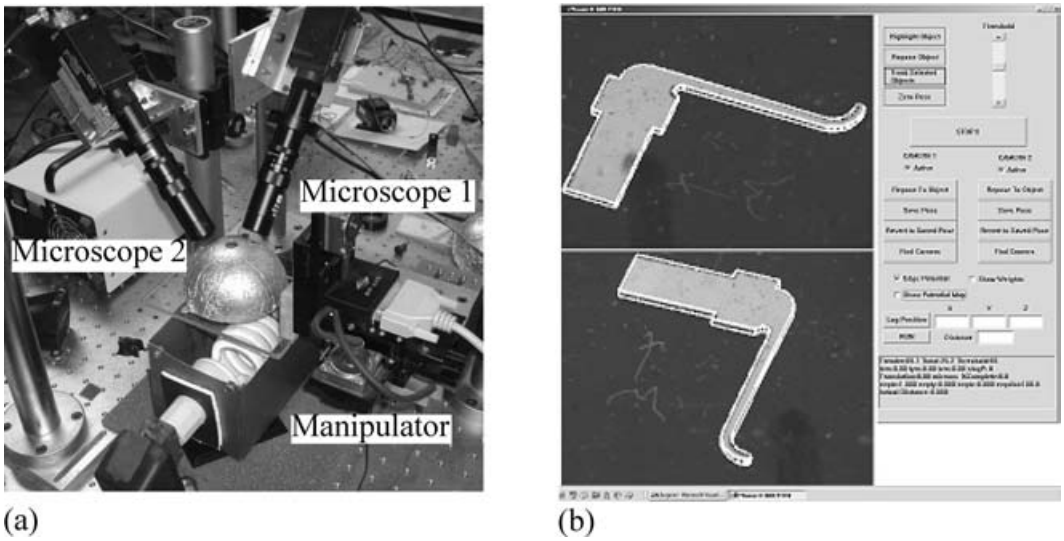

Fig. 8. (a) Experimental set-up (b) Screen capture of the visual tracking program.

Note that the 3D matching points $CP_i'^W$ were not measured by any form of range sensing but were estimated by the projection of model points CP_i^O onto the line-of sight vectors q_i. Therefore, after the best fitting ${R^*}_O^W$ and ${T^*}_O^W$ are found, a new set of $CP_i'^W$ are calculated using the same CP_i^O and q_i. The above steps for pose estimation are then repeated until a measure of convergence for ${R^*}_O^W$ and ${T^*}_O^W$ is satisfied.

3. EXPERIMENTAL RESULTS

3.1. Experimental set-up

Figure 8a shows the experimental set-up used to implement the CAD model based tracking method. Two cameras with variable zoom microscope lenses (Edmund Scientific VZM-300i) observe the MEMS components. These lenses were chosen for their large depth of field ($\sim$500 μm at 3X magnification). MEMS components are placed on a platform attached to a three axis micropositioning stage (Sutter MP-285). A reflective dome is placed over the transparent platform for diffuse illumination of the scene by the light source under the platform. A single PC (Pentium IV, 2.4 GHz) with two framegrabbers (Sensoray 611) is used to acquire and process the video images. The tracking software and the user interface were written in C++ and OpenGL. Figure 8b shows a screen capture of the program. The video images from both cameras are displayed along with CAD models superimposed on the objects. The system processes both images at 30 frames/sec.

Throughout the experiments, the spacing of control points on the model edges were set to 7 pixels and the search lines along the edge normals were set to 21 pixels length. These settings resulted in about 500 control points tracked by two cameras. The gradient threshold for edge detection was set

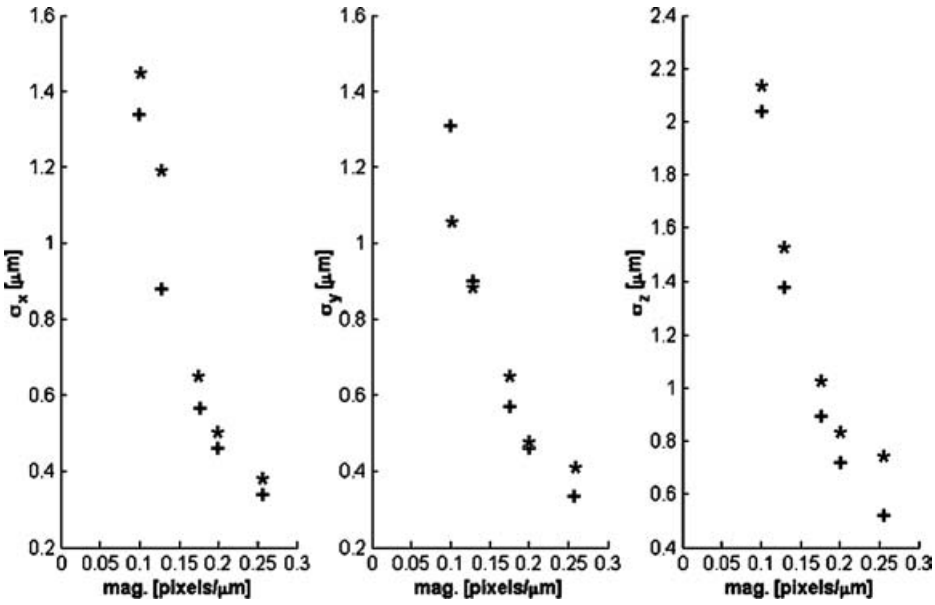

Fig. 9. Standard deviations of the position parameters along x, y, z axes with changing magnification. Measured values are shown by $*$, estimated values by $+$.

https://doi.org/10.1017/S0263574704000840 Published online by Cambridge University Press

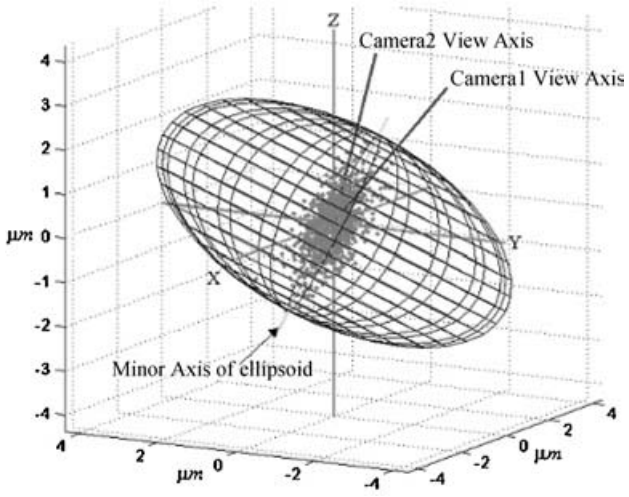

Fig. 10. Deviation of the measured position of a static object from the mean position.

to 40% of the difference between the highest and lowest intensities in each search line. These values were determined empirically and were found to be adequate for the single CPU system to achieve 30 frames/sec performance.

3.2. Precision

A set of experiments was performed to determine the precision of the visual tracking system. The precision of measurements is dependent on various factors including the overall geometry of the scene and the quality of the image. The combined geometrical factors, such as the magnification of the microscope-camera system, the size, shape and pose of the observed objects and the relative positions of the cameras, result in the final set of observed control points on the model edges. The uncertainty of the final pose measurement can also be related to noise in the image space measurements at each control point through the image Jacobian matrix J (2).

$$C_h = (J^T J)^{-1} J^T C_e ((J^T J)^{-1} J^T)^T \tag{16}$$

where C_h is the covariance matrix for the final estimated pose vector and C_e is the combined covariance matrix for the image space measurements. To test the precision of the system, a stationary, square shaped object 970 μm on a side was observed for 1000 frames under different magnifications of the zoom microscope lenses. The spacing of the control points was fixed at 7 pixels, therefore, the number of control points increases with increasing magnification. Figure 9 shows the observed and estimated 1σ deviation of the measured position vector at increasing magnifications selected within a practical range. The range of magnifications is limited by the object size in the image and depth-of-field of the camera lens system. The estimated values assume that the noise at each control point is independent from others and is normally distributed with $\sigma_{u,v} = 1.0$ pixel. An average precision of 1.0 μm was achieved. The estimated precision values are close to the observed values, showing that the system performance can be predicted using the CAD models. This ability is valuable for task/sensor planning. Figure 10 shows the collected data from one of the experiments along with the resolvability ellipsoid and the directions of the camera view axes. The deviations of the position vectors are largely spread along the shortest axis of the ellipsoid, as expected, since that is the direction of lowest resolvability.

3.3. Accuracy

To test the accuracy of the visual tracking system a square shaped object, 970 μm on a side, was translated by known amounts using the manipulator, and the measured displacement was compared to the commanded translation. The Sutter MP-285 manipulator is an open-loop device that uses a stepping motor in microstepping mode and a proprietary worm gear driven capstan drive along each axis

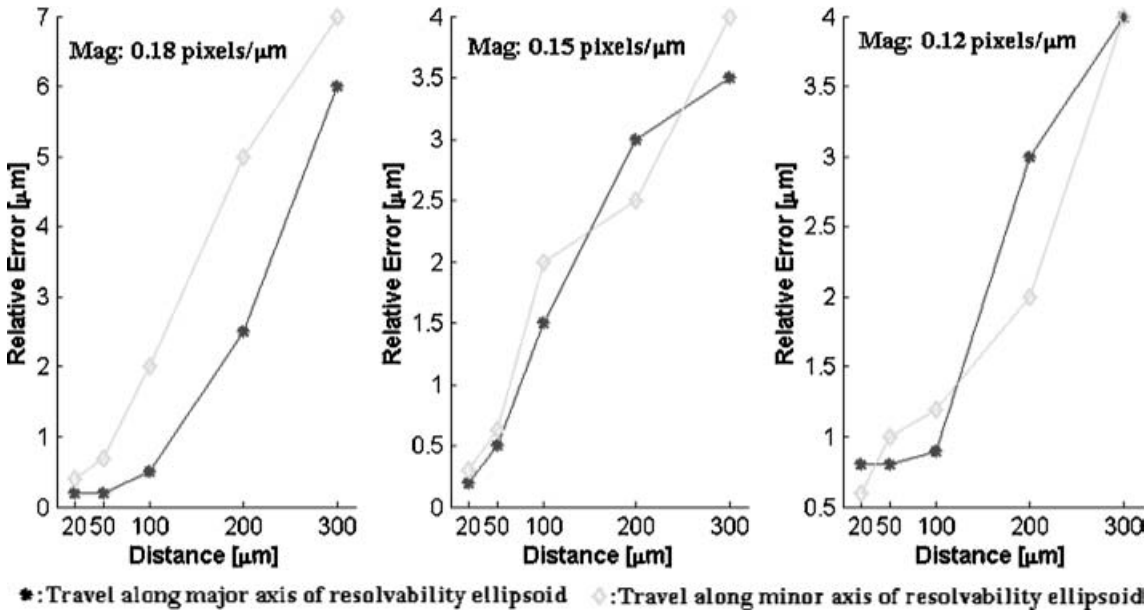

Fig. 11. Results of accuracy experiments.

https://doi.org/10.1017/S0263574704000840 Published online by Cambridge University Press

of motion. The command resolution of the manipulator is 0.040 μm. However, backlash and runout errors limit the accuracy. Backlash errors happen only when the drive axes change directions and are estimated to be "on the order of 1–2 μm" by the manufacturer. The manufacturer also estimated the runout errors for less than one revolution of the worm gear (corresponding to 800 μm travel along any axis) to be "less than 1% of travel." The travel distance during the accuracy tests was always less than 800 μm and "anti-backlash" moves were executed whenever the direction was reversed to eliminate the effect of backlash. To minimize the precision errors in measurements, the measured position vectors at the beginning and the end of the travel were averaged for 100 video frames. Each experiment consisted of 50 relative position measurements made after translations of 20, 50, 100, 200, and 300 μm along a fixed direction. Two directions of travel were used, one along the shortest axis of the resolvability ellipsoid, and the other along the longest axis. These steps were repeated for three different magnifications of the microscope lenses at 0.18, 0.15 and 0.12 pixels/μm. The results are plotted in Figure 11.

On the average, an accuracy better than 2% was achieved. The results for 0.18 pixels/magnification show a larger error for distances above 200 μm compared to smaller magnifications. At higher magnifications the object gets close to the depth of field limits of the lens after a large travel and part of the edges are effected from the reduced image sharpness increasing the errors. The measurements for translations along the minor and major axes of resolvability were not consistently better or worse than the other. This can be explained as the effect of averaging the measured position at the beginning and end of travel to minimize the precision errors which are directionally dependent.

4. CONCLUSIONS

A robust and cost effective microassembly process is essential for the development of advanced hybrid MEMS devices. While the difficulties of manipulation at the microscale make traditional open-loop manipulation unsuitable for microassembly, closed-loop control through visual feedback overcomes many of these difficulties and corrects for the positioning errors. A CAD model based approach to visual tracking is particularly suitable for flexible automation of microassembly.

The advantages of using CAD models for part/task description apply equally well to micro-scale visual servoing and common macro-scale automation tasks. However, the increased visual resolvability of the model based multi-camera tracking method becomes essential in micro-scale vision to overcome the insensitivity of microscope lenses along the view axis. Our experimental system has achieved 6-DOF tracking of MEMS components in real-time (30 Hz). A 1σ precision of 1.0 μm and a relative position accuracy of 2% was demonstrated.

The ability to achieve 6-DOF tracking in real-time makes the CAD model based method a viable alternative to the depth-from-defocus techniques commonly applied in micro-scale vision. Whereas the typically low depth of field of microscope lenses provide advantage for the depth-from-defocus method, the model based approach can still be used with high depth of field microscope lenses, as well as with other imaging methods such as SEM or Wavefront Coding[36] that provide increased depth of field in real-time.

References

1. H. Guckel, T. Earles, J. Klein, D. Zook and T. Ohnstein, "Electromagnetic Linear Actuators with Inductive Position Sensing for Micro Relay, Micro Valve and Precision Positioning Applications", *Transducers '95 Conference* (1995) **Vol. 1**, pp. 324–327.
2. Y. Zhou and B. J. Nelson, "Adhesion Force Modeling and Measurement for Micromanipulation," *Microrobotics and Micromanipulation* (A. Sulzmann and B. J. Nelson, Editors) *Proceedings of SPIE* (1998) Vol. 3519, pp. 169–180.
3. M. Mukundarishnan and B. J. Nelson, "Micropart Feature Design For Visually Servoed Microassembly", *IEEE Int. Conf. on Robotics and Automation* (2000) pp. 965–970.
4. J. T. Feddema and R. W. Simon, "CAD-driven Microassembly and Visual Servoing", *IEEE Int. Conf. on Robotics and Automation* (1998) **Vol. 2**, pp. 1212–1219.
5. M. Chen, S. Kume, A. Rizzi and R. Hollis, "Visually Guided Coordination for Distributed Precision Assembly", *IEEE Int. Conf. on Robotics and Automation* (2000) **Vol. 2**, pp. 1651–1656.
6. B. Vikramaditya and B. J. Nelson, "Visually Guided Microassembly Using Optical Microscopes and Active Vision Techniques", *IEEE Int. Conf. on Robotics and Automation* (1997) pp. 3172–3177.
7. A. Buerkle and S. Fatikow, "Laser Measuring System for a Flexible Microrobot-based Micromanipulation Station", *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS 2000)* (2000) **Vol. 1**, pp. 799–804.
8. A. Sulzmann, J. Breguet and J. Jacot, "Micromotor Assembly Using High Accurate Optical Vision Feedback for Microrobot Relative 3D Displacement in Submicron Range", *IEEE Int. Conf. on Solid State Sensors and Actuators* (1997) **Vol. 1**, pp. 279–282.
9. S. Fatikow and U. Rembold, "An Automated Microrobot-based Desktop Station for Micro Assembly and Handling of Micro-objects", *IEEE Conference on Emerging Technologies and Factory Automation* (1996) **Vol. 2**, pp. 586–592.
10. K. Koyano and T. Sato, "Micro Object Handling System with Concentrated Visual Fields and New Handling Skills", *IEEE Int. Conf. on Robotics and Automation* (1996) pp. 2541–2548.
11. T. Sato, K. Koyano, M. Nakao and Y. Hatamura, "Novel Manipulator for Micro Object Handling as Interface Between Micro and Human Worlds", *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS 1993)* (1993) Vol. 3, pp. 1674–1681.
12. T. Sato, T. Kameya, H. Miyazaki and Y. Hatamura, "Hand-Eye System in Nano Manipulation World", *IEEE Int. Conf. on Robotics and Automation* (1995) pp. 59–66.
13. A. Burkle, F. Schmoeckel, H. Worn, B. P. Amavasai, F. Caparrelli and J. R. Travis, "A Versatile Vision System for Micromanipulation Tasks", *Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems (MFI 2001)* (2001) pp. 271–276.
14. S. J. Lee, K. Kim, D. Kim, J. Park and G. Park, "Recognizing and Tracking of 3D-shaped Micro Parts Using Multiple Visions for Micromanipulation", *Int. Symp. on Micromechatronics and Human Science* (2001) pp. 203–210.
15. R. Hollis and J. Gowdy, "Miniature Factories for Precision Assembly", *International Workshop on Microfactories*, Tsukuba, Japan (1998) pp. 9–14.
16. J. T. Feddema, C. S. G. Lee and O. R. Mitchell, "Weighted Selection of Image Features for Resolved Rate Visual Feedback Control", *IEEE Transactions on Robotics & Automation* **1**, No. 1, 31–47 (1991).

https://doi.org/10.1017/S0263574704000840 Published online by Cambridge University Press

17. J. Shi and C. Tomasi, "Good Features to Track", *IEEE Conference on Computer Vision and Pattern Recognition* (1994) pp. 593–600.
18. F. Janabi-Sharifi and W. J. Wilson, "Automatic Selection of Image Features for Visual Servoing", *IEEE Transactions on Robotics and Automation* **13**, No. 6, 890–903 (1997).
19. N. Papanikolopoulos, P. K. Khosla and T. Kanade, "Vision and Control Techniques for Robotic Visual Tracking", *IEEE Int. Conf. on Robotics and Automation* (1991) pp. 857–864.
20. S. Allegro, C. Chanel and J. Jacot, "Autofocus for Automated Microassembly Under a Microscope", *IEEE Int. Conf. on Image Processing* (1996) **Vol. 1**, pp. 677–680.
21. D. Lowe, "Fitting Parameterized Three-Dimensional Models to Images", *IEEE Transactions on Pattern Analysis and Machine Intelligence* **13**, No. 5, 441–450 (1991).
22. P. Wunsch and G. Hirzinger, "Registration of CAD-models to Images by Iterative Inverse Perspective Matching", *Int. Conf. on Pattern Recognition* (1996) pp. 78–83.
23. J. Stavnitzky and D. Capson, "Multiple Camera Model-Based 3-D Visual Servo", *IEEE Transactions on Robotics & Automation* **16**, No. 6, 732–739 (2000).
24. M. Tonko and H. Nagel, "Model-Based Stereo-Tracking of Non-Polyhedral Objects for Automatic Disassembly Experiments", *Int. Journal of Computer Vision* **37**, No. 1, 99–118 (2000).
25. J.-C. Latombe, *Robot Motion Planning* (Kluwer Academic Publishers, 1991).
26. B. J. Nelson and P. K. Khosla, "Vision Resolvability for Visually Servoed Manipulation", *Journal of Robotic Systems* **13**, No. 2, 75–93 (February, 1996).
27. J. D. Foley, A. van Dam, S. Feiner and J. Hughes, *Computer Graphics: Principles and Practice* (Addison-Wesley, 2nd ed. in C, 1996).
28. R. Carceroni and C. Brown, "Numerical Methods for Model Based Pose Recovery", *Technical Report 659* (Computer Science Dept., University of Rochester, August 1997).
29. C. P. Lu, G. D. Hager and E. Mjolsness, "Fast and Globally Convergent Pose Estimation from Video Images", *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**, No. 6, 610–622 (2000).
30. K. Kanatani, "Analysis of 3-D Rotation Fitting", *IEEE Transactions on Pattern Analysis and Machine Intelligence* **16**, No. 5, 543–549 (1994).
31. B. K. Horn, "Closed-from Solution of Absolute Orientation Using Unit Quarternions", *J. of Opt. Soc. of America* **A-4**, 629–642 (1987).
32. M. W. Walker, L. Shao and R. A. Volz, "Estimating 3-D Location Parameters Using Dual Number Quarternions", *CVGIP: Image Understanding* **54**, No. 3, 358–367 (1991).
33. B. K. Horn, H. M. Hilden and S. Negahdaripour, "Closed-from Solution of Absolute Orientation Using Orthonormal Matrices", *J. of Opt. Soc. of America* **A-5**, 1128–1135 (1988).
34. K. Arun, T. Huang and S. Blostein, "Least-squares Fitting of Two 3-D Point Sets", *IEEE Transactions on Pattern Analysis and Machine Intelligence* **9**, 698–700 (1987).
35. A. Lorusso, D. Eggert and R. B. Fisher, "A Comparison of Four Algorithms for Estimating 3-D Rigid Transformations", *Proc. British Machine Vision Conference BMVC95* (1995) pp. 237–246.
36. E. R. Dowski and W. T. Cathey, "Extended Depth of Field Through Wavefront Coding", *Applied Optics* **34**, No. 11, 1859–1866 (1995).

https://doi.org/10.1017/S0263574704000840 Published online by Cambridge University Press