

Alignment Using an Uncalibrated Camera System

Billibon H. Yoshimi, *Member, IEEE*, and Peter K. Allen, *Member, IEEE*

Abstract—We describe a method for the visual control of a robotic system which does not require the formulation of an explicit calibration between image coordinates and the world coordinates. Calibration is known to be a difficult and error prone process. By extracting control information directly from the image, we free our technique from the errors normally associated with a fixed calibration. We attach a camera system to a robot such that the camera system and the robot's gripper rotate simultaneously. As the camera system rotates about the gripper's rotational axis, the circular path traced out by a point-like feature projects to an elliptical path in image space. We gather the projected feature points over part of a rotation and fit the gathered data to an ellipse. The distance from the rotational axis to the feature point in world space is proportional to the size of the generated ellipse. As the rotational axis gets closer to the feature, the feature's projected path will form smaller and smaller ellipses. When the rotational axis is directly above the object, the trajectory degenerates from an ellipse to a single point. We demonstrate the efficacy of the algorithm on the peg-in-hole problem. In this scenario, the peg is aligned with the rotational axis of the end effector and the task is to align the peg with the hole. The algorithm uses an approximation to the Image Jacobian to control the movement of the robot.

I. INTRODUCTION

IN many assembly tasks, there exists a need for alignment and precision insertion of parts. This usually involves understanding the 3-D relationship between the parts to be aligned. Machine vision systems are capable of recovering these relationships and can be used as the basis for controlling alignment and insertion tasks. A common approach to solving this problem using visual control is based upon calibrating the vision system (*image space*) with the 3-D environment (*world space*). The success of this approach depends upon an extremely accurate calibration. Current calibration techniques [1] model both the geometric and optical parameters of camera/lens systems, and can provide highly accurate calibration. However, these methods are often difficult to understand and inconvenient to use in many robotic environments. They are usually described by several nonlinear equations, and require sometimes painstaking effort to use. Another problem is that calibrated systems are typically accurate only over a small subspace of the robot's workspace. As the robot leaves the region where the calibration occurred, accuracy degenerates quickly, thus requiring that the system be recalibrated. Finally, calibration also becomes difficult when the camera systems are dynamic and move in the environment.

Manuscript received August 6, 1993; revised April 20, 1994. This work was supported in part by DARPA contract DACA-76-92-C-007, NSF grants IRI-86-57151, and CDA-90-24735.

The authors are with The Center for Research in Intelligent Systems, Department of Computer Science, Columbia University, New York, NY 10027 USA.

IEEE Log Number 9409726.

This paper describes a method that can be used to control a robot using an uncalibrated camera system. Because the camera requires no calibration, it can be reconfigured easily for use in different task settings. In our method, images taken from a camera rotating about an axis of a robot can be used as the basis of a control algorithm to accurately position the robot. As the camera system rotates, a tracked feature point projects to an elliptical path in image space. The distance from the rotational axis to the feature point in world space is proportional to the size of the generated ellipse. As the rotational axis gets closer to the feature, the feature's projected path will form smaller and smaller ellipses. When the rotational axis is directly above the object, the trajectory degenerates from an ellipse to a single point. We have demonstrated this method on a classic robot-control problem: how to insert a peg into a hole using only vision to control the peg's movement. Our method allows a robot system to move to a location based purely on a tracked feature's change in image coordinates.

Our work is characteristic of research currently examined in the domain of active vision, the use of vision to control the feedback of a servoing loop. Some representative works in this area include: Skaar *et al.*'s work with camera space manipulations [2], Sarachik's work with mobile robot calibration [3], Tsuji's work with monitoring mobile robot environments [4], Papanikopolous *et al.*'s work in developing new controllers for visual servoing problems [5], Hollinghurst and Cipolla's work in the loosely calibrated visual control of a gripper system [6], and Castaño and Hutchinson's work with visual constraint planes [7]. It also ties in with the recent spate of work in self-calibration and the use of uncalibrated systems [8], [9]. The effect of rotational epipolar motion, the key component in the feedback loop of our system, was first investigated by Bolles *et al.* [10]. This effect has been used in several scenarios ranging from Raviv [11] and others who have used it to constrain camera fixation to Kutulakos and Dyer [12] who used it to reconstruct curved surfaces to Cipolla and Blake [13] who applied it to determine time to impact. For a comprehensive review of recent work in the field of active vision and visual servoing see [14], [15].

II. OVERVIEW OF METHOD

Research in peg-in-hole tasks is rich and varied. Some of the classic techniques include the work of Nevins and Whitney [16] in Remote-Center-Compliance (RCC) and the work of Lozano-Pérez *et al.* [17] with back-projections. The method we propose concentrates on the task of aligning the end effector with the target. Our work is extremely useful in solving the initial positioning problem for alignment and is consistent with the use of the previously mentioned methods.

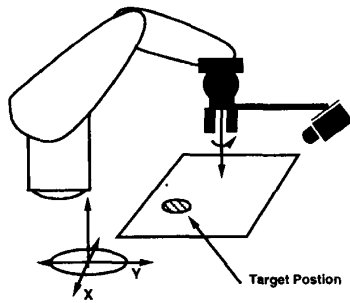


Fig. 1. Experimental setup.



Fig. 2. Overhead view of camera, robot, and multiple target setup.



Fig. 3. View of targets from camera.

A simplified schematic of our experimental setup is shown in Figs. 1 and 2 is an overhead view. The task of the vision-robot system is to insert the tip of the probe, which is mounted on the end-effector of the robot arm coincident with the robot's final axis, into one of the small holes in the block on the table in Fig. 3 (the hole is selected in the image by the operator). Fig. 3 is the vision system's view of the real world (the vision system is located to the left of the probe in Fig. 2). The camera is mounted on the wrist of the end-effector such that it rotates about the final axis of the robot and images the area directly below the end-effector.

The method relies on the following simple effect: as we rotate the camera system about the robot's final axis, those

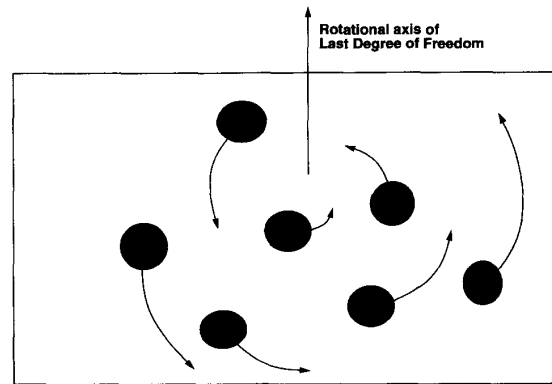


Fig. 4. Image space movement of objects due to camera rotation about the final axis of the robot.

objects which are farther away from the axis of rotation move a greater distance in the image plane than those objects closer to the axis (see Fig. 4). If we track these objects in the image plane during the rotation, they trace out elliptical paths. We refer to these ellipses as image-ellipses. If the only movement in the robot-camera system is the rotation, the object will trace out a conic section, an ellipse under certain conditions. The degenerate conditions are:

- if the rotational axis is aligned with the optical axis—a circle,
- if the object is already aligned to the rotational axis—a point,
- if the rotational axis is perpendicular to the optical axis—a line,
- if, during a rotation, the object intersects the plane containing the focal point parallel to the image plane and intersects the plane at only one point during the complete rotation—a parabola, and
- if, during a rotation, the object passes through the plane containing the focal point parallel to the image plane (hence intersecting the plane at two points)—a hyperbola.

As long as the path swept out by the object does not pass through the plane containing the focal point parallel to the imaging plane, its projection will be an ellipse.

Fig. 5 shows a series of simulated image-ellipses formed by tracking a feature point in the image plane. The family of image-ellipses was generated by changing the distance of the tracked feature point from the axis about which the camera system rotates. The center dot is formed when the tracked feature point is coincident with the rotational axis. As the tracked feature point is moved further and further away from the rotational axis, it sweeps out larger and larger image-ellipses. Note that the image ellipses swept out are not confocal.

In Fig. 6, the robot-camera system is constrained to move in the plane A which is defined by the circle swept by the camera around the rotational axis. The camera subsystem itself is constrained to move in a circular orbit around the axis of the last joint of the robot. In order to align the peg with the hole, we examine the movement of a hole feature, which projects

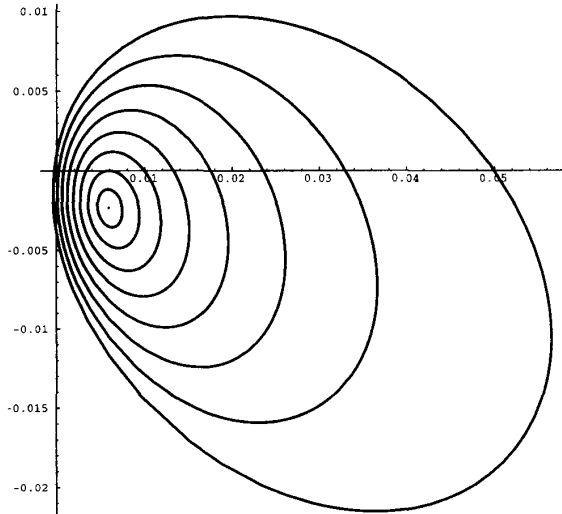


Fig. 5. Simulated family of image-ellipses generated by changing the distance of the tracked feature point from the axis about which the camera system rotates. Real image-ellipses can be viewed in Fig. 8.

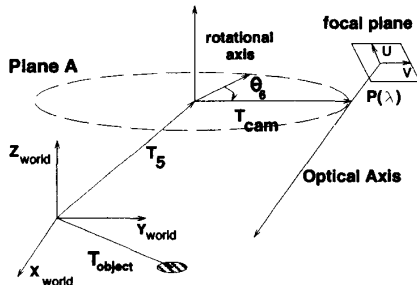


Fig. 6. Overview of the coordinate systems.

to a small set of pixels in image space. By moving the robot-camera system to different positions in plane *A* and rotating the camera system about the final axis, we can generate image-ellipses and compute their areas. The problem now reduces to one of finding a search strategy which minimizes the number of positions examined before the alignment condition occurs. We broke the peg-in-hole task into two parts: the alignment task and the actual insertion task. The alignment task moves the end effector in a plane in robot space until the alignment condition occurs (when the peg and hole lie on the same axis). Once the alignment has been performed in plane *A*, the only movement necessary for insertion of the peg into the hole is a pure translation along the rotational axis (in our scenario, the *Z*-axis).

We define our experimental setup (see Fig. 6) using the following set of parameters (we also indicate which quantities are unknown at the beginning of the experiment):

- $(X_{world}, Y_{world}, Z_{world})$ are the world coordinate axes.
- T_5 , the transform from the world coordinate system to the end-effector of the robot, minus the last rotational degree of freedom, is known.
- R , the rotational transform for the final axis, θ_6 , of the robot is known.

- T_{cam} , the camera transform matrix, is unknown.
- T_{object} , the transform from the world system to the feature point where the probe is to be inserted, is unknown.
- (U, V) are the image coordinate axes.
- $P(\lambda)$, the perspective effect introduced by the camera system with focal length λ is unknown.

We also make the following assumptions about our experiments:

- Once the camera is mounted, it remains fixed with respect to the rotational axis of the final joint (6th) of the robot.
- The camera must be able to image the target object during any movement. The object should never leave the focal plane.
- The object remains a fixed distance below the plane containing the end effector (the workspace of the robot movements). Tracking is performed in this 2D space.
- The features to be tracked are point-like features.

Intuitively, all we require is that the robot knows the location of its end-effector and that the camera system rotates about the robot's final axis while keeping the feature point in view. Note that we do not know where the camera is located with respect to the robot and we do not require the optical axis of the camera to intersect the final rotational axis of the robot.

III. ON THE RECOVERY OF IMAGE-ELLIPSE PARAMETERS

Many methods exist for recovering of image ellipses from sampled point data. Two recent works in this area are Safaei-Rad *et al.* [18] and Sawhney [19]. These algorithms typically use data sampled from the full circumference of the ellipse (i.e., sampled over 2π radians). While these methods work very well, sampling over a complete rotation about the robot's axis is very slow. Our algorithm does not require that we sample the ellipse over a complete rotation. In our current setup, we sweep our camera system over $\pi/2$ radians in world coordinates.¹

As we sweep the camera system about the rotational axis, we collect sets of samples (θ_i, U_i, V_i) where θ_i is the rotational angle of the robot's final axis as the tracked feature point is imaged, and (U_i, V_i) are the coordinates of the feature point in camera space. We then parameterize the curve traced out by the feature as

$$U(\theta) = A \cos(\theta) + B \sin(\theta) + C \quad (1)$$

$$V(\theta) = D \cos(\theta) + E \sin(\theta) + F. \quad (2)$$

We fit these equations to the triplet data using a least squares technique. Upon recovering the ellipse parameters, we are able to compute the area enclosed by the ellipse by using the following formula:

$$Area = (A^2 + B^2 + D^2 + E^2) \frac{\pi}{2}. \quad (3)$$

¹The $\pi/2$ radian swath in world coordinates does not usually correspond to an equal $\pi/2$ radian swath in camera space. Because of perspective distortion, the path swept out in the camera system will either be smaller or larger than $\pi/2$ radians. While this should affect the accuracy of our results, the method we propose appear to handle the small amounts of phase distortion robustly.

(which was derived using the previous two equations and Green's Theorem). The full proof that the parametric curves generated by these equations are ellipses is contained in [20] along with a discussion of why other methods for recovering image-ellipse parameters were not used.

IV. THE IMAGE JACOBIAN

Our search in the space of image ellipses for the alignment condition requires us to move the robot based upon the size of the imaged ellipse. We compute the required world space movement by using an approximation to the Image Jacobian. Image Jacobians have been used by a number of other researchers including Feddema et al. [21], Hashimoto et al. in [15], and Castaño and Hutchinson [7]. These methods use the Jacobian to map image-based, feature movements to Cartesian displacements. The idea behind the Image Jacobian is to model the differential relationship between the camera system and the robotic control system in order to accurately predict the effects of small changes in one system on the other. It is a local, linear approximation to this relationship.

We can model the relationship between the camera and world space in the following way:

$$w \begin{bmatrix} U_i \\ V_i \\ 1 \end{bmatrix} = P(\lambda) \cdot [T_5 \cdot R(\theta_6) \cdot T_{cam}]^{-1} \cdot \begin{bmatrix} X_{world} \\ Y_{world} \\ Z_{world} \\ 1 \end{bmatrix}. \quad (4)$$

Equation (4) can be differentiated with respect to time and the resulting equation is the differential relationship which relates the two coordinate systems. However, full knowledge of (4) is necessary to correctly derive the Jacobian matrix. In our system, since we assume that the calibration between the camera system and robot system is unknown, the task of recovering the actual Image Jacobian cannot be done directly.

We are only interested in control movements in the 2-D X - Y plane of the robot (we assume the alignment occurs at a fixed depth Z above the object). Accordingly, we can state the differential relationship between the camera coordinate system and the robot control system as follows:

$$\begin{bmatrix} \delta X \\ \delta Y \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} \delta U \\ \delta V \end{bmatrix} \quad (5)$$

which means that small perturbations around some point in image space (U, V) can be linked to some small, linear move in world space (X, Y). Espiau *et al.* [22] have recently shown that convergence is guaranteed if the signs of the perturbations are correct and that performance increases as the approximation approaches the ideal value. The variables a, b, c , and d (the components of the Jacobian) are dependent on the robot parameters, the transformation between the camera system and the robot system, and the camera system parameters. We calculate the Image Jacobian by making two moves in world space, observing the movements of the feature in image space, and inserting these values into (5) to find a, b, c and d . By empirically calculating the Image Jacobian at each new point (and throwing away the information from previous calculations), we can use the new estimates to move the robot

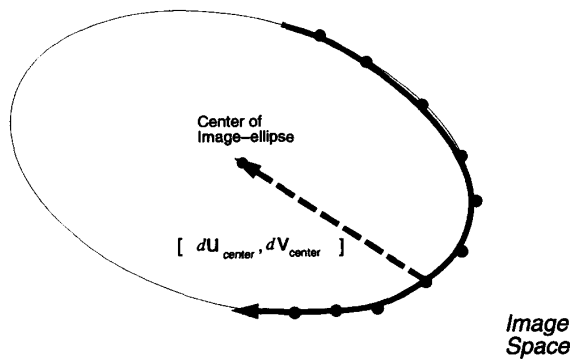


Fig. 7. Calculation of image space movement ($\delta U_{center}, \delta V_{center}$). The movement of the robot to the alignment position is estimated by transforming this vector with the Image Jacobian.

to the correct alignment position even though we have not calibrated the two systems. Our method is:

- 1) The robot moves to some real world position (X, Y, Z).
- 2) We rotate the camera $\pi/2$ radians about the robot's final axis, tracking the feature point in image space and we calculate the area of the image-ellipse traced out.
- 3) At the midpoint of the rotation, $\pi/4$ radians, the robot freezes the rotation angle and translates to positions ($X + \delta X, Y, Z$) and ($X, Y + \delta Y, Z$). The change in image coordinates of the tracked object feature ($\delta U, \delta V$) is calculated. Each change in X - Y coordinates ($\delta X, \delta Y$) results in a change in image coordinates ($\delta U, \delta V$) which we substitute in for the components of the Image Jacobian, a, b, c , and d , in (5).
- 4) The alignment condition occurs when the image-ellipse formed by tracking the object feature's path has zero area. We estimate the alignment position in image space as the center of the image-ellipse (see Fig. 7). We can compute the vector ($\delta U_{center}, \delta V_{center}$) which is the vector from the position in the image where we calculated the Jacobian to the center of the projected ellipse. We can now use our computed Image Jacobian to transform the vector ($\delta U_{center}, \delta V_{center}$) into a control movement in Cartesian space ($\delta X, \delta Y$).
- 5) This procedure is repeated until the error signal results in either a single point or an image-ellipse of area $\leq 1.0 \text{ pixel}^2$.

This algorithm controls the robot by estimating differential changes at each step. Although the actual computed movement is larger than a differential move, the vector ($\delta U_{center}, \delta V_{center}$) provides both an accurate direction of movement in image space as well as a reasonable magnitude of movement so that we may limit overshoot and oscillation.

V. EXPERIMENTAL RESULTS

In Fig. 1, we show a schematic of the system set up used for testing the new alignment method. We mounted a Sony XC-77 CCD camera in a bracket off the end effector of a Puma 560 robot. The camera was not calibrated or position-constrained when initially placed. The images were digitized at 256 x 242 resolution and 8 b gray scale at standard NTSC

TABLE I

Positions examined by algorithm			
Position Number	X (in mm)	Y (in mm)	Ellipse Area
1	0.0	0.0	17880.0
2	-84.517176	-13.628160	271.0
3	-76.971801	-29.850992	25.0
4	-80.212628	-25.240072	.78

frame rates using the PIPE parallel image processing engine. The object was positioned so the robot would not encounter singularities when moving to the new control positions. The other constraints on our system are listed in Section VI.

Once the robot moved to its initial X - Y position in plane A , the user indicated the position in the image of the object feature to be tracked. The vision algorithm then created a moving edge detection region centered at that point. This moving region was used to track the feature point as the camera rotated about the final axis of the robot. The camera rotated about the robot's final axis at approximately .035 rad/s. The speed was limited by the digitizing speed of the framebuffer and displaying the output on a host computer; we are currently implementing a much faster version that takes advantage of concurrent image processing and robot movement.

We started the robot system at point (0, 0) in Cartesian space with the object approximately 100 mm away in the X - Y plane. We ran the experiment using the Image Jacobian to control the movement of the robot system. The tracked feature was a 2 mm diameter hole. The robot system was able to place a point-like peg (a tapered probe) within 3mm of the center of the hole using uncalibrated camera data. In addition, the algorithm performed the same task over several different runs with about the same accuracy. Table I shows sample data taken from one of the experiments, Fig. 8 shows an overlay of all ellipses generated by the Image Jacobian control algorithm while trying to find the alignment condition, and Fig. 9 shows that the positioning moves in the robot world coordinate system. Notice that the system converges rapidly to the solution state.

VI. CONCLUSIONS

We have described a technique for performing an alignment task using uncalibrated cameras. The technique exploits a simple geometric effect, rotational invariance, to perform the alignment. In our system, we mount a camera on the wrist of an end-effector such that it rotates about the final axis of a robot and images the area directly below the end-effector. By tracking the movement of a point-like feature in image space, we can determine that a feature is aligned with the final axis when its projection simply rotates and does not translate during the rotation.

We implemented an Image Jacobian-based control to find the alignment condition. The key idea behind the Image

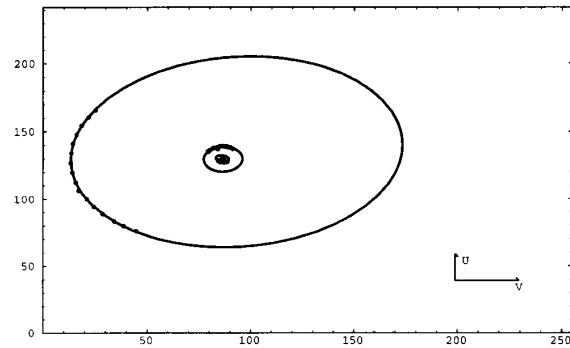


Fig. 8. The image-ellipses formed by tracking the object over all four positions moved to by the robot.

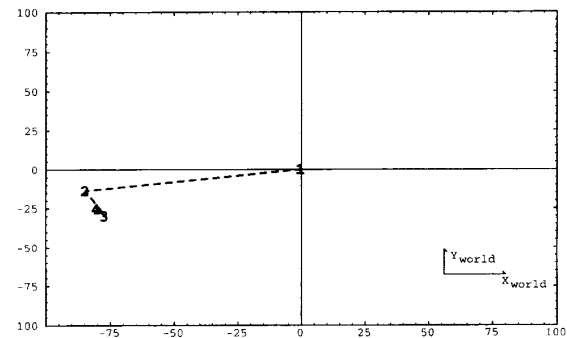


Fig. 9. Robot positions moved to during the search for the alignment condition.

Jacobian was to move to the center of the image-ellipse (the best approximation to the point of rotational invariance) using a discrete approximation of the Jacobian at a given point. It was possible to create an algorithm which converged quickly to the alignment position. Using this controller, we were able to make a robot system place a point-like peg within 3 mm of the center of a 2 mm hole using an uncalibrated camera with a high degree of repeatability.

This research indicates the possible existence of a whole class of servoing techniques which are orthogonal to calibration-oriented methods. These techniques use simple geometric constraints to give robots the ability to servo visually. While calibration is necessary and useful for many vision-robot tasks, we have shown that it is possible to use a noncalibrated technique to perform a task which was originally thought to require calibrated vision. Also, by not using a calibrated camera, we do not suffer from the known problems associated with calibrated systems.

Our future research includes relaxing the constant Z constraint, the implementation of a faster image-space tracking system, and the investigation of other noncalibrated techniques for visual-servoing.

REFERENCES

- [1] R. Y. Tsai, "Synopsis of recent progress on camera calibration for 3D machine vision," *Robot. Rev.*, 1989.
- [2] S. Skaar, W. Brockman, and R. Hanson, "Camera-space manipulation," *Int. J. Robot. Res.*, vol. 6, no. 4, pp. 20-32, 1987.

- [3] K. Sarachik, "Characterizing an indoor environment with a mobile robot and uncalibrated stereo," in *Proc. IEEE Conf. Robot. and Automat.*, 1989, pp. 984-989.
- [4] S. Tsuji, "Monitoring of a building environment by a mobile robot," in *Robot. Rev.*, 1985, vol. 2, pp. 349-356.
- [5] N. Papanikolopoulos, P. Khosla, and T. Kanade, "Vision and control techniques for robotic visual tracking," in *IEEE Int. Conf. Robot. and Automat.*, 1991, vol. 1, pp. 857-864.
- [6] N. Hollinghurst and R. Cipolla, "Uncalibrated stereo hand-eye coordination," Tech. Rep. CUED/F-INFENG/TR126, Dept. Engineering, Univ. Cambridge, 1993.
- [7] A. Castano and S. Hutchinson, "Visual compliance: Task-directed visual servo control," Tech. Rep. UIUC-BI-AI-RCV-93-01, The Beckman Inst., Univ. Illinois, 1993.
- [8] S. Maybank and O. Faugeras, "A theory of self-calibration of a moving camera," *Int. J. Comput. Vision*, vol. 8, no. 3, pp. 123-151, 1992.
- [9] R. Hartley, "Estimation of relative camera positions for uncalibrated cameras," in *Proc. 2nd Europe. Conf. Comput. Vision (ECCV-92)*, 1993, pp. 579-587.
- [10] R. C. Bolles, H. H. Baker, and D. H. Marimon, "Epipolar-plane image analysis: An approach to determining structure from motion," *Int. J. Comput. Vision*, vol. 1, pp. 7-55, 1987.
- [11] D. Raviv, "A quantitative approach to camera fixation," Tech. Rep. NISTIR 4324, NIST-Robot Systems Division-Sensory Intell. Group, Gaithersburg, MD, May 1990.
- [12] K. N. Kutulakos and Charles R. Dyer, "Recovering shape by purposive viewpoint adjustment," in *Proc. 1991 Conf. Comput. Vision and Patt. Recognition*, 1991, pp. 16-22.
- [13] R. Cipolla and A. Blake, "The dynamic analysis of apparent contours," in *1991 Stockholm Workshop MCV*, 1991, pp. 616-623.
- [14] A. Blake and A. Yuille, Eds., *Active Vision*. Cambridge, MA: MIT Press, 1992.
- [15] K. Hashimoto, Ed., *Visual Servoing*. New York: World Scientific, 1993.
- [16] S. Nevins and D. Whitney, "Computer-controlled assembly," *Sci. Amer.*, vol. 238, pp. 62-74, Feb. 1978.
- [17] T. Lozano-Perez, M. Mason, and R. Taylor, "Automatic synthesis of fine-motion strategies for robots," *Int. J. Robot. Res.*, vol. 3, no. 1, pp. 3-24, 1984.
- [18] R. Saffae-Rad, I. Tchoukanov, K. C. Smith, and B. Benhabib, "Three-dimensional location estimation of circular features for machine vision," *IEEE Trans. Robot. and Automat.*, vol. 8, pp. 624-640, 1992.
- [19] H. S. Sawhney, "Spatial and temporal grouping in the interpretation of image motion," Ph.D. dissertation, Univ. Mass.-Amherst, Feb 1992.
- [20] B. H. Yoshimi and P. Allen, "Alignment using an uncalibrated camera system," Tech. Rep. TR-002-93, Columbia Univ., NY, 1993.
- [21] J. Feddema, C. S. G. Lee, and O. Mitchell, "Weighted selection of image features for resolved rate visual feedback control," *IEEE Trans. Robot. and Automat.*, vol. 7, pp. 31-47, Feb. 1991.
- [22] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Trans. Robot. and Automat.*, vol. 8, pp. 313-326, 1992.

Billibon H. Yoshimi (M'95) received the B.S. and M.S. degrees in computer science from Columbia University and is currently pursuing the Doctorate degree.

He is a graduate research assistant working under Peter Allen in Computer Science at Columbia University. His current research interests include visually guided robot servoing, real-time robotics, and real-time/parallel vision algorithms.

While at Columbia, he has received several awards including the ARVO-National Eye Institute Travel Award, the NCR Stakeholders Award, Columbia SEAS Bronze Key Award and an Outstanding Student Award.

Peter K. Allen (M'85) received the A.B. degree from Brown University in mathematics-economics, the M.S. degree in computer science from the University of Oregon, and the Ph.D. degree in computer science from the University of Pennsylvania.

He is Associate Professor of Computer Science at Columbia University and Director of the Center for Research in Intelligent Systems. His current research interests include real-time computer vision, using dextrous robotic hands for object recognition and task-level manipulation, and model-based sensor planning.

Dr. Allen was the recipient of the CBS Foundation Fellowship, Army Research Office fellowship and the Rubinoff Award for innovative uses of computers. In recognition of his work, he has been named a Presidential Young Investigator by the National Science Foundation. He is a member of ACM and AAAI.