Uncalibrated Stereo Hand–Eye Coordination

Nicholas Hollinghurst and Roberto Cipolla
Department of Engineering,
University of Cambridge,
Cambridge CB2 1PZ, England.

Abstract

This paper describes a system that combines stereo vision with a 5-DOF robotic manipulator, to enable it to locate and reach for objects by sight.

Our system uses an *affine stereo* algorithm, a simple but robust approximation to the geometry of stereo vision, to estimate positions and surface orientations. It can be calibrated very easily with just four reference points. These are defined by the robot itself, moving the gripper to four known positions (*self-calibration*).

The inevitable small errors are corrected by a feedback mechanism which implements *image-based control* of the gripper's position and orientation. Integral to this feedback mechanism is the use of *affine active contour models* which track the real-time motion of the gripper across the two images.

Experiments show the system to be remarkably immune to unexpected translations and rotations of the cameras and changes of focal length — even after it has 'calibrated' itself.

A future goal is to use affine stereo to implement shape-based grasp planning, enabling the robot to pick up a wide range of unidentified objects left in its workspace. At present it can only pick up simple wooden blocks and tracks a single planar contour on its target object.

1 Introduction

When humans grasp and manipulate objects, they almost invariably do so with the aid of vision. Visual information is used to locate and identify things, and to decide how (and if) they should be grasped. Visual feedback helps us guide our hands around obstacles and align them accurately with their goal. *Hand–Eye Coordination* gives us a flexibility and dexterity of movement that no machine can match.

Vision systems for robotics usually need to be *calibrated* — the camera geometry must be measured to a high degree of accuracy. A well-calibrated stereo rig can accurately determine the positions of things to be grasped [10]. However, if calibration is erroneous or the cameras are disturbed, the system will fail gracelessly.

An alternative approach in hand–eye applications where a manipulator moves to a visually-specified target, is to use visual feedback to match manipulator and target positions *in the image*. Exact spatial coordinates are not required, and a well-chosen feedback architecture can correct for quite serious inaccuracies in camera calibration (as well as inaccurate kinematic modelling) [13].

Here we demonstrate the use of a *weak perspective* model of stereo vision which, though of modest accuracy, is robust to camera disturbances and is easy to calibrate. In fact, the system calibrates itself automatically whenever it is initialised by observing the robot's gripper moving to four reference points.

Closed-loop control is achieved by tracking the gripper's movements across the two images to estimate its position and orientation relative to the target object. This is done with a form of *active contour model* resembling a B-spline snake [3] but constrained to deform only affinely.

2 Weak Perspective and Affine Stereo

2.1 Assumption of weak perspective

The full perspective transformation between world and image coordinates is conventionally analysed using the *pinhole camera* model, in which image-plane coordinates (u, v) are ratios of world coordinates (x_c, y_c, z_c) in a camera-centred frame, thus: $u = fx_c/z_c,\ v = fy_c/z_c$. The relation between the camera-centred and some other world coordinate frame consists of *rotation* ($\mathbf{R}$) and *translation* ($\mathbf{t}$) components representing the camera's orientation and position. Using homogeneous coordinates (with scale factor s for convenience),

$$\begin{bmatrix} su \\ sv \\ sf \end{bmatrix} = \begin{bmatrix} \mathrm{r}_{11} & \mathrm{r}_{12} & \mathrm{r}_{13} & \mathrm{t}_1 \\ \mathrm{r}_{21} & \mathrm{r}_{22} & \mathrm{r}_{23} & \mathrm{t}_2 \\ \mathrm{r}_{31} & \mathrm{r}_{32} & \mathrm{r}_{33} & \mathrm{t}_3 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}. \tag{1}$$

Consider a camera viewing a compact scene of interest from distance h. For convenience, we can translate the whole system so that the scene lies close to the world origin. t_3, the component of $\mathbf{t}$ along the optical axis, will then equal h. As distance increases relative to the radius of the scene, z_c/h will tend to unity for all points, and the projection becomes approximately linear:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{f}{h} \begin{bmatrix} \mathrm{r}_{11} & \mathrm{r}_{12} & \mathrm{r}_{13} & \mathrm{t}_1 \\ \mathrm{r}_{21} & \mathrm{r}_{22} & \mathrm{r}_{23} & \mathrm{t}_2 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}. \tag{2}$$

This formulation assumes that images are not distorted by variations in depth, and is known as *weak perspective* [9]. It is an orthographic projection scaled by a factor inversely proportional to h. It can be shown that this assumption results in an error which is, at worst, $\Delta h/h$ times the scene's image size. Where small objects are viewed from two or three metres distance, as in many practical vision applications, the assumption of weak perspective is reasonable.

The relation between *image-plane coordinates* (u, v) and *pixel addresses* (X, Y) can be modelled by an affine transformation (to represent offsets, scaling and shearing), and the entire projection written very simply as a linear mapping:

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} \mathrm{p}_{11} & \mathrm{p}_{12} & \mathrm{p}_{13} & \mathrm{p}_{14} \\ \mathrm{p}_{21} & \mathrm{p}_{22} & \mathrm{p}_{23} & \mathrm{p}_{24} \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}. \tag{3}$$

The eight coefficients p_{ij} efficiently represent all intrinsic and extrinsic camera parameters [11]. This simple approximation to the projection transformation will be used as the camera model throughout the paper.

2.2 Motion of planar objects in weak perspective

There are many situations in computer vision where an object must be *tracked* as its image moves across a view. Here we consider the simple, but not uncommon, case where the object is small and planar.

We can define a coordinate system centred about the object itself so that it lies within the xy plane. If the object is small compared to the camera distance, we again have weak perspective, and a special case of (3):

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} \mathrm{a}_{11} & \mathrm{a}_{12} & \mathrm{a}_{13} \\ \mathrm{a}_{21} & \mathrm{a}_{22} & \mathrm{a}_{23} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}. \tag{4}$$

We see that the transformation from a plane in the world to the image plane is affine. As the camera moves relative to the object, parameters a_{ij} will change and the image will undergo translation, rotation, divergence and deformation, but remain *affine-invariant* (figure 1).

This is a powerful constraint that can be exploited when tracking a planar object. It tells us that the shape of the image will deform only affinely as the object moves, and that there will exist an affine transformation between any two views of the same plane.

2.3 Affine stereo

Correspondence and the epipolar constraint

Under the weak perspective assumption, the two image coordinates of each point are a linear projection of its world coordinates. To recover all three world coordinates it is obvious that two (or more) views are needed. It is then necessary to identify which features in the different views correspond to the same point in space, a task known as the *Correspondence Problem.*

The image positions of a world feature in two images are not independent, but are related by an *epipolar constraint.* In weak perspective stereo, this takes the form of a single linear constraint among the four image coordinates. Most correspondence algorithms exploit this constraint, which reduces the search for matching features to a single dimension.

The affine stereo formulation

We assume that the cameras do not move relative to the scene during each period of use. Combining information from a pair images, we have four image coordinates (X, Y, X', Y') for each point, all linear functions of the three world coordinates (x_w, y_w, z_w):

$$\begin{bmatrix} X \\ Y \\ X' \\ Y' \end{bmatrix} = \begin{bmatrix} & \mathbf{P} & \\ & & \\ & \mathbf{P}' & \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}. \tag{5}$$

Here $[\mathbf{P}\mathbf{P}']^T$ is a 4×4 matrix, formed from the p_{ij} coefficients of (3) for the two cameras. To calibrate the system it is necessary to observe a minimum of four non-coplanar *reference points*, yielding sixteen simultaneous linear equations from which $[\mathbf{P}\mathbf{P}']^T$ may be found.

Calibration is better conditioned than with full-perspective stereo, because the system has fewer parameters and is amenable to solution by linear techniques (full projective stereo can be represented by 24 linear coefficients but there are nonlinear constraints on those coefficients [4]). With noisy image data, greater accuracy may be obtained by observing more than four points, using a recursive linear estimator.

Once the coefficients are known, world coordinates can be obtained by inverting (5), using a least-squares method to resolve the redundant information. Errors in calibration will manifest themselves as a linear distortion of the perceived coordinate frame.

It is *not* essential to calibrate a stereo vision system to obtain useful 3-D information about the world. Instead, four of the points observed may be given arbitrary world coordinates (such as $(0,0,0)$, $(0,0,1)$, $(0,1,0)$ and $(1,0,0)$). The appropriate solution for $[\mathbf{PP}']^T$ will define a coordinate frame which is an arbitrary 3-D affine transformation of the 'true' Cartesian frame, preserving affine shape properties such as collinearity and coplanarity. This is in accordance with Koenderink and van Doorn's *Affine Structure-from-Motion Theorem* [8].

In hand–eye applications, it might instead be convenient to calibrate the vision system in the coordinate space in which the manipulator is controlled (assuming this maps approximately linearly to Cartesian coordinates).

Recovery of surface orientation in affine stereo

Any two views of the same planar surface will be affine-equivalent, and there will exist an affine transformation that maps one image to the other. This transformation can be used to recover surface orientation [2]. Surface orientation in space is most conveniently represented by a surface normal vector $\mathbf{n}$. We can obtain it by the vector product of two non-collinear vectors in the plane.

Consider the standard unit vectors $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$ in one image and suppose they were the projections of some vectors on the object surface. If the linear mapping between images is represented by a 2×3 matrix $\mathbf{A}$, then the first two columns of $\mathbf{A}$ itself will be the corresponding vectors in the other image. As the centroid of the plane will map to both image centroids, we can easily use it and the above pairs of vectors to find three points in space on the plane (by inverting $[\mathbf{PP}']^T$) and hence the surface orientation.

3 Tracking using Affine Active Contours

An *Active Contour* (or *'Snake'*) [7] is a curve defined in the image plane that moves and deforms according to various 'forces'. These include *external forces*, which are local properties of the image, and *internal forces* which are functions of the snake's own shape. Typically, a snake will be attracted to maxima of image intensity gradient, and used to track the edges of a moving object.

3.1 Anatomy

Our model-based trackers are a novel form of active contour. They resemble B-spline snakes [3] but consist of (in the order of 100) discrete sampling points, rather than a smooth curve [6]. We use them to track planar surfaces bounded by contours, on the robot gripper and the object to be grasped. Pairs of trackers operate independently in the two stereo views. The trackers can deform only affinely, to track planes viewed in weak perspective [1]. This constraint leads to a more efficient and reliable tracker than a B-spline snake, that is less easily confused by background contours or partial occlusion.

Each tracker is a model of the image shape it is tracking, with sampling points at regular intervals around the edge. At each sampling point there is a *local edge-finder* which measures the offset between modelled and actual edge positions in the image, by searching for the maximum of gradient along a short line segment [5]. Due to the so-called *aperture problem* [12], only the normal component of this offset can be recovered at any point (figure 2).

The positions of the sampling points are expressed in affine coordinates, and their image positions depend upon the tracker's *local origin* and two *basis vectors*. These are described by six parameters, which change over time as the object is tracked. The contour tangent directions at each point are also described in terms of the basis vectors.

3.2 Algorithm

At each time-step the tracker moves and deforms to minimise the sum of squares of offsets between model and image edges. In our implementation this is done in two stages. First the optimal translation is found, then the deformation, rotation, divergence components are calculated. Splitting the task into these two stages was found to increase stability, as fewer parameters were being estimated at once. To find the optimal translation $\mathbf{t}$ to account for normal offset h_i at each sampling point whose normal direction is $\mathbf{n}_i$, we solve the following equation:

$$h_i = \mathbf{n}_i \cdot \mathbf{t} + \epsilon_i. \tag{6}$$

ϵ_i is the error term, and we solve the whole system of equations using a least-squares method to minimise $\sum \epsilon_i^2$.

Once the translation has been calculated, the other components are estimated. It is assumed that the distortion is centred about the tracker's local origin (normally its centroid, to optimally decouple it from translation). The effects of translation $(\mathbf{n}_i \cdot \mathbf{t})$ are subtracted from each normal offset, leaving a residual offset. We can then find the matrix $\mathbf{A}$ that maps image coordinates to displacement.

$$(h_i - \mathbf{n}_i \cdot \mathbf{t}) = \mathbf{n}_i \cdot (\mathbf{A}\mathbf{p}_i) + \epsilon'_i, \tag{7}$$

where $\mathbf{p}_i$ is the sampling point's position relative to the local origin and ϵ'_i is again the error term to be minimised.

In practice this formulation leads to trackers that are as willing to distort as to translate, which can cause problems when the tracked surface moves whilst partially obscured. We therefore use a simplified version of this equation that ignores the aperture problem (equating the normal component with the whole displacement):

$$(h_i - \mathbf{n}_i \cdot \mathbf{t})\mathbf{n}_i = \mathbf{A}\mathbf{p}_i + \mathbf{E}_i. \tag{8}$$

$\mathbf{E}_i$ is a vector, and our implementation solves the equations to minimise $\sum |\mathbf{E}_i|^2$. This produces a more stable tracker that, although sluggish to deform, is well suited to those practical tracking tasks where motion is dominated by the translation component. The tracker positions are updated from $\mathbf{t}$ and $\mathbf{A}$ using a first-order predictive filter. This enhances performance when tracking rapidly-moving objects.

4 Visual Feedback for Hand–Eye Coordination

Affine stereo is a simplified stereo vision formulation that is very easily calibrated. Conversely, it is of rather low accuracy. Nevertheless, it gives reliable *qualitative* information about the relative positions of points and can, of course, indicate when they are in precisely the same place. We therefore use a feedback control mechanism to help to guide the gripper to the target, using affine stereo to compute the relative position and orientation of their respective tracked surfaces.

Since the reference points used to self-calibrate are specified in the *controller's* coordinate space, linear errors in the kinematic model are effectively bypassed. The system must still cope with any nonlinearities in control, as well as those caused by strong perspective effects.

An integral feedback control architecture is employed. The feedback term is the difference between the vectors that describe the position and orientation of the target and gripper, as seen by the vision system. This term is integrated by summing at each time step, and the resulting vector used to position the robot (figure 3).

The manipulator moves in discrete steps, through a distance proportional to the difference between the gripper's perceived coordinates and those of the target plane. The gain is below unity to prevent ringing or instability, even when the vision system is miscalibrated. This process is repeated until the perceived positions of the two coincide (or, for grasping, we can introduce a fixed offset).

5 Implementation and Experiments

5.1 Equipment

The system is implemented on a Sun SPARCstation 10 with a Data Cell S2200 frame grabber. The manipulator is a Scorbot ER-7 robot arm, which has 5 degrees of freedom and a parallel-jawed gripper. The robot has its own 68000-based controller which implements the low-level control loop and provides a Cartesian kinematic model. Images are obtained from two inexpensive CCD cameras placed 2m–3m from the robot's workspace. The angle between the cameras is in the range of 15–30 degrees.

5.2 Implementation

When the system is started up, it begins by opening and closing the jaws of the gripper. By observing the image difference, it is able to locate the gripper and set up a pair of affine trackers as instances of a 2-D template. The trackers will then follow the gripper's movements continuously. Tracking can be implemented on the Sun at frame rate. The robot moves to four preset points to calibrate the system in terms of the controller's coordinate space.

A target object is found by similar means — observing the image changes when it is placed in the manipulator's workspace. Alternatively it may be selected from a monitor screen using the mouse. There is no pre-defined model of the target shape, so a pair of 'exploding' B-spline snakes [3] are used to locate the contours delimiting the 'target surface' in the two images. The snakes are then converted to a pair of affine trackers. The two trackers are made affine-equivalent so that the surface orientation can be recovered easily from their basis vectors. The target surface is then tracked, in case it is moved, or to compensate for camera motions.

The orientation of the gripper of a 5-DOF manipulator is constrained by its 'missing' axis, and this constraint changes continuously as it moves. To avoid this problem, the present implementation keeps the gripper vertical, reducing the number of degrees of freedom to four. Its orientation is then described by a single *roll angle*. It is assumed that the target plane is also vertical.

By introducing modifications and offsets to the feedback mechanism (which would otherwise try to superimpose the gripper and the target), two 'behaviours' have been implemented. The *grasping behaviour* causes the gripper to approach the target from above (to avoid collisions) with the gripper turned through an angle of 90 degrees, to grasp it normal to its visible surface. The *tracking behaviour* causes it to follow the target continuously, hovering a few centimetres above it (figure 5).

5.3 Results

Without feedback control, the robot locates its target only approximately (typically to within 5cm in a 50cm workspace). With a feedback gain of 0.75 the gripper converges on its target in three or four control iterations. If the system is not disturbed it will take a straight-line path. The system has so far demonstrated its robustness by continuing to track and grasp objects despite:

Kinematic errors. Linear offsets or scalings of the controller's coordinate system are absorbed by the self-calibration process with complete transparency. Slight non-linear distortions to the kinematics are corrected for by the visual feedback loop, though large errors introduce a risk of ringing and instability unless the gain is reduced.

Camera disturbances. The system continues to function when its cameras are subjected to small translations, rotations and zooms, even after it has self-calibrated. Large disturbances to camera geometry cause the gripper to take a curved path towards the target, and require more control iterations to get there.

Strong perspective. The condition of weak perspective throughout the workspace does not seem to be essential for image-based control and the system can function when the cameras are as close as 1.5 metres (the robot's reach is a little under 1 metre). However the feedback gain must be reduced to below 0.5, or the system will overshoot on motions towards the cameras.

Figure 5 shows four frames from a tracking sequence (all taken through the same camera). The cameras are about two metres from the workspace. Tracking of position and orientation is maintained even when one of the cameras is rotated about its optical axis and zoomed.

6 Future Developments

The current system is based upon matching the positions and orientations of two *planes*. Since the gripper and target planes must be visible in both images at all times, neither can rotate through more than about 120 degrees. We intend to develop the system to track the gripper using a three-dimensional rigid model [6], drawing information from both images simultaneously. We also aim to identify and track more than one of the surfaces of the object to be grasped, both for 3-D tracking and also for analysis of its 3-D shape.

We plan to equip such a system with a grasp planner that uses the relative positions, sizes and orientations of the visible surfaces on the object, to direct the robot to grasp unmodelled objects in a suitable way.

7 Conclusion

Affine stereo provides a robust interpretation of image position and disparity that degrades gracefully when cameras are disturbed. It is suitable for uncalibrated and self-calibrating systems.

By defining the working coordinate system in terms of the robot's abilities, linear errors in its kinematics are bypassed. The remaining non-linearities can be handled using image-based control and feedback. We have shown that this can be achieved cheaply and effectively using a novel form of snake to track planar features on the gripper and target.

Such a system has been implemented and found to be highly robust, without unduly sacrificing performance (in terms of speed to converge on the target).

Acknowledgements

The authors gratefully acknowledge the donation of a robot manipulator by the Olivetti Research Lab. Cambridge, and the financial support of SERC.

References

[1] A. Blake, R. Curwen, and A. Zisserman. Affine-invariant contour tracking with automatic control of spatiotemporal scale. In *Proc. 4th Int. Conf. on Computer Vision*, 1993.

[2] R. Cipolla and A. Blake. Surface orientation and time to contact from image divergence and deformation. In G. Sandini, editor, *Proc. 2nd European Conference on Computer Vision*, pages 187–202. Springer–Verlag, 1992.

[3] R. Cipolla and A. Blake. Surface shape from the deformation of apparent contours. *Int. Journal of Computer Vision*, 9(2):83–112, 1992.

[4] R. Cipolla, Y. Okamoto, and Y. Kuno. Robust structure from motion using motion parallax. In *Proc. 4th Int. Conf. on Computer Vision*, 1993.

[5] R. Curwen and A. Blake. Dynamic contours: real-time active splines. In A. Blake and A. Yuille, editors, *Active Vision*. MIT Press, 1992.

[6] C. Harris. Tracking with rigid models. In A. Blake and A. Yuille, editors, *Active Vision*. MIT Press, 1992.

[7] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: active contour models. In *Proc. 1st Int. Conf. on Computer Vision*, pages 259–268, 1987.

[8] J.J. Koenderink and A.J. van Doorn. Affine structure from motion. *J. Opt. Soc. America*, pages 377–385, 1991.

[9] L.G. Roberts. Machine perception of three-dimensional solids. In J.T. Tippet, editor, *Optical and Electro-optical Information Processing*. MIT Press, 1965.

[10] M. Rygol, S. Pollard, and C. Brown. A multiprocessor 3d vision system for pick-and-place. In *British Mach. Vision Conf.*, 1990.

[11] R.Y. Tsai. An efficient and accurate camera calibration technique for 3D machine vision. In *Proc IEEE CVPR 86*, 1986.

[12] S. Ullman. *The interpretation of visual motion.* MIT Press, Cambridge,USA, 1979.

[13] D.F.H. Wolfe, S.W. Wijesoma, and R.J. Richards. Eye-to-hand coordination for vision-guided robot pick-and-place operations. In *J. Adv. Manuf. Eng*, 1990.

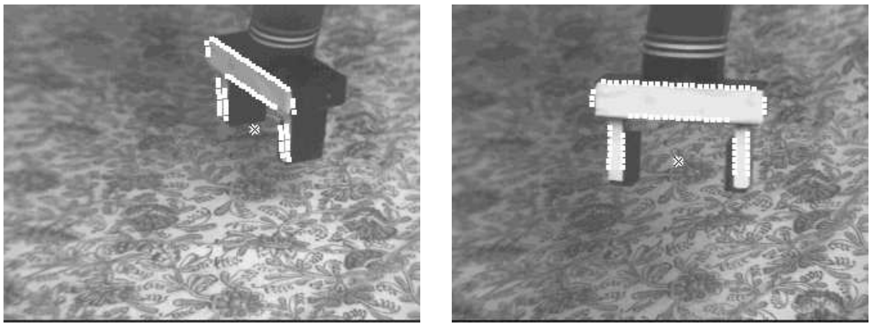

Figure 1: The gripper being tracked as it translates and rotates in weak perspective. The origin and sampling points of the tracker are shown in white. The front of the gripper is approximately planar, and its image shape remains affine-invariant.

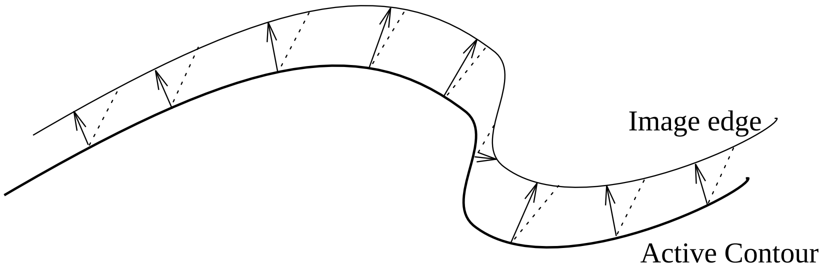

Figure 2: An active contour: an edge-finder searches normal to the contour at each sampling point (arrows). Only the normal component of the offsets can be recovered locally (the aperture problem). The optimal translation (dotted) can only be found globally.

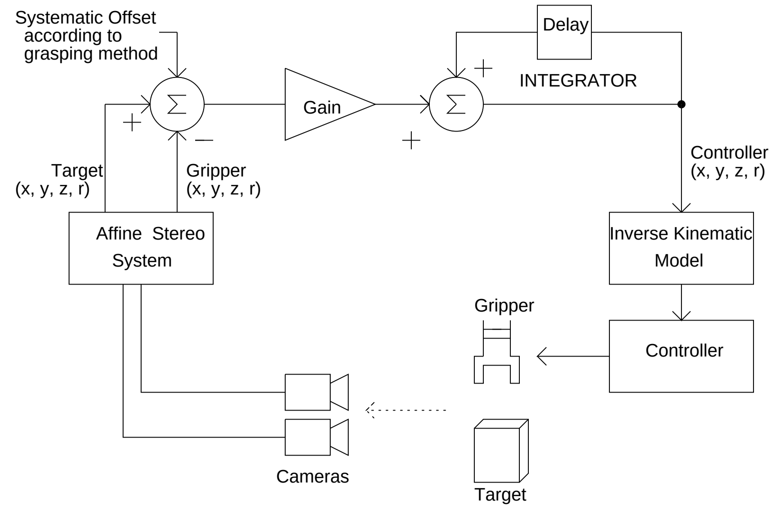

Figure 3: The control structure of the system, showing the use of visual feedback.

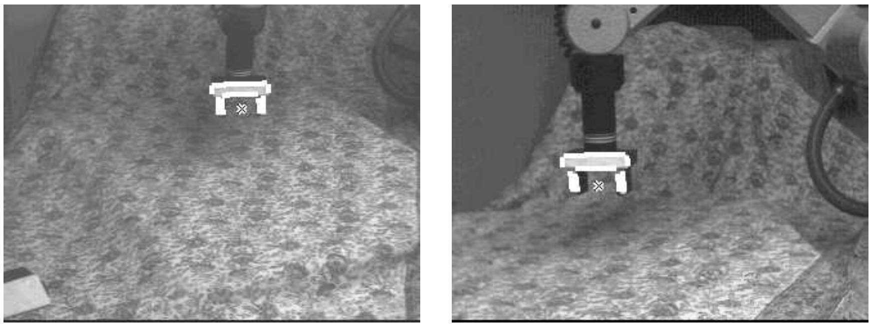

Figure 4: A stereo pair showing the robot gripper at one of the four reference points used for calibration. Active contour models are overlaid in white.

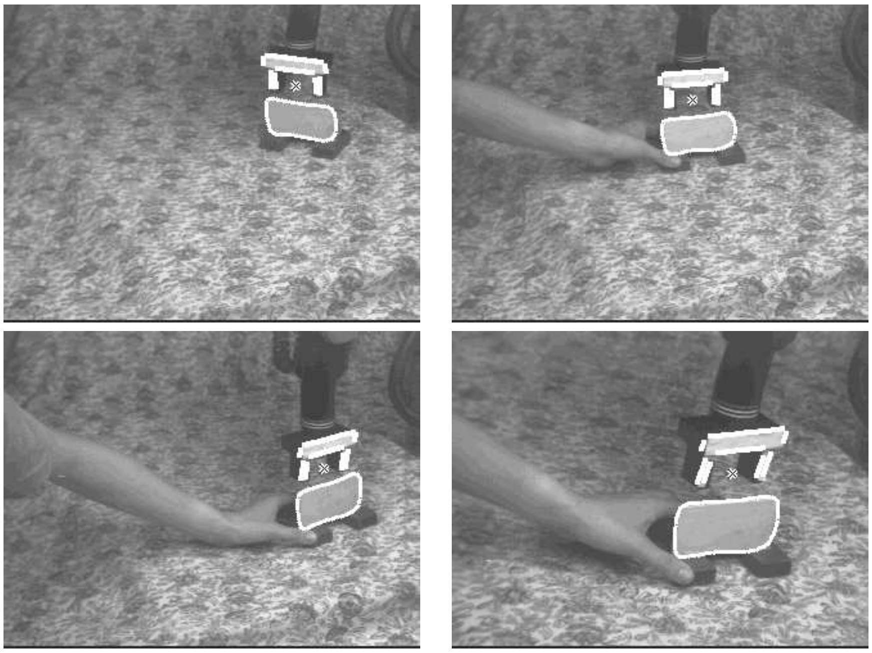

Figure 5: The robot is tracking its quarry, guided by the position and orientation of the target contour (view through left camera). On the target surface is an *affine snake* — an affine tracker obtained by 'exploding' a B-spline snake from the centre of the object. A slight offset has been introduced into the control loop to cause the gripper to hover above it. Last frame: one of the cameras has been rotated and zoomed, but the system continues to operate successfully with visual feedback.