Servoing Mechanisms for Peg-In-Hole Assembly Operations

Josef Pauli, Arne Schmidt, and Gerald Sommer

Christian-Albrechts-Universität zu Kiel
Institut für Informatik und Praktische Mathematik
Preußerstraße 1–9, D-24105 Kiel, Germany
www.ks.informatik.uni-kiel.de/~jpa

Abstract. Image-based effector servoing is a process of perception-action cycles for handling a robot effector under continual visual feedback. Apart from the primary goal of manipulating objects we apply servoing mechanisms also for determining camera features, e.g. the optical axes of cameras, and for actively changing the view, e.g. for inspecting the object shape. A peg-in-hole application is treated by a 6-DOF manipulator and a stereo camera head. The two robot components are mounted on separate platforms and can be steered independently. In the first phase (inspection phase), the robot hand carries an object into the field of view of one camera, then approaches the object along the optical axis to the camera, rotates the object for reaching an optimal view, and finally inspects the object shape in detail. In the second phase (insertion phase), the system localizes a board containing holes of different shapes, determines the relevant hole based on the extracted object shape, then approaches the object, and finally inserts it into the hole. At present, the robot system has the competence to handle cylindrical and cuboid pegs. For treating more complicated objects the system must be extended with more sophisticated strategies for the inspection and/or insertion phase.

1 Introduction

Image-based robot servoing (visual servoing) is the backbone of Robot Vision systems. The book edited by Hashimoto [3] collects various approaches of automatic control of mechanical systems using visual sensory feedback. A tutorial introduction to visual servo control of robotic manipulators has been published by Hutchinson et al. [5]. Quite recently, a special issue of the International Journal on Computer Vision has been devoted to image-based robot servoing [4].

Our work demonstrates the usefulness of servoing for treating all sub-tasks involved in an overall robotic application. The novelty is to consider servoing as a universal mechanism for *camera-robot calibration*, *active viewing*, *shape inspection*, and *object manipulation*. Furthermore, we consider minimalism principles by extracting just the necessary image information and *avoiding 3D reconstruction*, which leads to real-time usage. Related to the particular application of *peg-in-hole assembly operations* it is favourable to integrate video and force information [6]. Due to limited paper size, we concentrate on servoing mechanisms for the vision-related sub-tasks of the overall peg-in-hole application.

As a survey, we describe (in Section 2) the components of the robot system, present (3) the general measurement-based control mechanism, use (4) servoing for determining the optical camera axis, apply (5) the servoing mechanism for optimally viewing and inspecting the object (Figure 1), and use (in Sections 6 and 7) servoing to suitably approach the object to the relevant hole (Figure 2).

2 System description

The computer system consists of a Sun Enterprise (E4000 with 4 UltraSparc processors) for image processing and of special processors for computing the inverse kinematics and motor signals. The robot system is composed of a robot manipulator including a hand with parallel jaw fingers and a robot head including two monochrome, stereo cameras. Based on six rotational joints of the manipulator the robot hand can be moved in arbitrary position and orientation within a certain working space. Additionally, there is a linear joint at the robot hand for opening and closing the two fingers. The tool center point is defined at the position of the hand tip, which is fixed in the middle point between the two finger tips. The robot head is mounted on a mobile platform and is equipped with motorized pan, tilt, and vergence degrees-of-freedom (DOF). Additionally, the stereo camera has motorized zooming and focusing facilities.

Fig. 1. Robot head, manipulator; approach an object towards a camera for shape inspection.

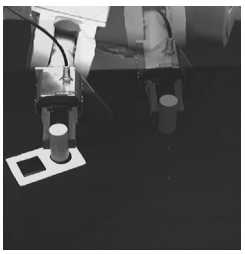

Fig. 2. Vision-based approaching a cylindrical peg to a circular hole.

3 Mechanism of measurement-based control

The robot system is characterized by a *fixed state vector* S^c which is inherent constant in the system, and by a *variable state vector* $S^v(t)$ which can be changed through a *vector of control signals* $C(t)$ at time t. State and control vector are specified in the manipulator coordinate system. A subsequent state vector $S^v(t+1)$ is obtained by a transition function f^{ts}, e.g. addition of $S^v(t)$ and $C(t)$.

$$S^v(t+1) := f^{ts}(S^v(t), C(t)) \tag{1}$$

A *control function* f^{ct} is responsible for generating the control vector $C(t)$. It is based on the current state vector $S^v(t)$, a *current measurement vector* $Q(t)$ and a *desired measurement vector* Q^*.

$$C(t) := f^{ct}(S^v(t), Q(t), Q^*) \tag{2}$$

A *measurement function* f^{ms} is responsible for taking and analyzing images, and thereof generating the current and desired measurement vectors $Q(t)$ and Q^*. They are represented in the coordinate systems of the cameras.

$$Q(t) := f^{ms}(S^v(t), S^c) \tag{3}$$

Control function f^{ct} describes the relation between changes in different coordinate systems, e.g. $S^v(t)$ in the manipulator and $Q(t)$ in the image coordinate system. For defining this function, the Jacobian will be computed for a projection matrix $\mathcal{M}$ which lineary approximates (in projective spaces) the relation between the manipulator coordinate system and the image coordinate system.

$$\mathcal{M} := \begin{pmatrix} \mathcal{M}_1^v \\ \mathcal{M}_2^v \\ \mathcal{M}_3^v \end{pmatrix} \quad ; \quad with \quad \begin{array}{l} \mathcal{M}_1^v := (m_{11}, m_{12}, m_{13}, m_{14}) \\ \mathcal{M}_2^v := (m_{21}, m_{22}, m_{23}, m_{24}) \\ \mathcal{M}_3^v := (m_{31}, m_{32}, m_{33}, m_{34}) \end{array} \tag{4}$$

The usage of the projection matrix is specified according to [2, pp. 55-58]. Given a point in homogeneous manipulator coordinates $P := (X, Y, Z, 1)^T$, the position in homogeneous image coordinates $p := (x, y, 1)^T$ can be obtained.

$$p := f^{pr}(P) := \begin{pmatrix} f_1^{pr}(P) \\ f_2^{pr}(P) \\ f_3^{pr}(P) \end{pmatrix} := \frac{1}{\xi} \cdot \mathcal{M} \cdot P \quad ; \quad with \quad \xi := \mathcal{M}_3^v \cdot P \tag{5}$$

Matrix $\mathcal{M}$ is determined with simple linear methods by considering the training samples of corresponding 3D points and 2D points. The scalar parameters m_{ij} represent a combination of extrinsic and intrinsic camera parameters which we leave implicit. The specific definition of normalizing factor ξ in equation (5) guarantees that function $f_3^{pr}(P)$ is constant 1, i.e. the homogeneous image coordinates of position p are in normalized form. The *Jacobian* $\mathcal{J}$ for the transformation f^{pr} in equation (5), i.e. for projection matrix $\mathcal{M}$, is computed as follows.

$$\mathcal{J}(P) := \begin{pmatrix} \frac{\partial f_1^{pr}}{\partial X}(P) & \frac{\partial f_1^{pr}}{\partial Y}(P) & \frac{\partial f_1^{pr}}{\partial Z}(P) \\ \frac{\partial f_2^{pr}}{\partial X}(P) & \frac{\partial f_2^{pr}}{\partial Y}(P) & \frac{\partial f_2^{pr}}{\partial Z}(P) \end{pmatrix} := \begin{pmatrix} \frac{m_{11} \cdot \mathcal{M}_3^v \cdot P - m_{31} \cdot \mathcal{M}_1^v \cdot P}{(\mathcal{M}_3^v \cdot P) \cdot (\mathcal{M}_3^v \cdot P)} & \cdots \\ \vdots & \ddots \end{pmatrix} \tag{6}$$

Control function f^{ct} is based on deviations between current and desired image measurements and should generate changes in manipulator coordinates. For this purpose, the pseudo-inverse of the Jacobian is relevant (see following sections).

4 Servoing for estimating optical axes

For estimating the *optical axis of a camera* relative to the basis coordinate system of the manipulator we apply image-based hand-effector servoing. The optical axis intersects the image plane approximately at the center. By servoing the hand-effector such that the two-dimensional projection of the hand tip reaches the image center, we finally obtain a 3D position which is a point on the optical axis,

approximately. By applying this procedure at two different distances from the camera one obtains two distinct points located (approximately) on the optical axis which are used for its estimation. Two virtual planes are specified which are parallel to the $(\boldsymbol{Y}, \boldsymbol{Z})$ plane with constant offsets X_1 and X_2 on the $\boldsymbol{X}$-axis. The movement of the hand-effector is restricted just to these planes (see Figure 3). Accordingly, the generic definition of the Jacobian $\mathcal{J}$ in equation (6) can be restricted to the second and third columns, because the coordinates on the $\boldsymbol{X}$-axis are fixed. A quadratic Jacobian matrix is obtained (with two rows and columns) which must be inverted, i.e. $\mathcal{J}^{\dagger}(P) := \mathcal{J}^{-1}(P)$.

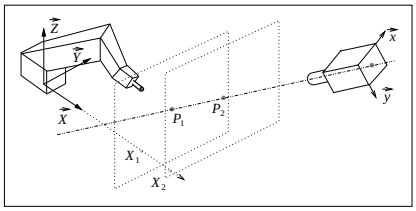

Fig. 3. Hand-effector servoing for estimating the optical axis of a camera.

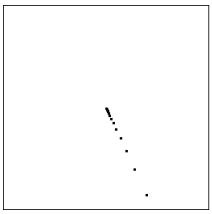

Fig. 4. Course of detected hand tip towards image center.

The current measurement vector $Q(t)$ is defined as the 2D image location of the hand tip and the desired measurement vector Q^* as the image center point. Hough transformation and normalized cross correlation are used in combination for detecting the hand tip in the images. The variable state vector $S^v(t)$ consists of the two variable coordinates of the tool center point in the selected plane $(X_1, \boldsymbol{Y}, \boldsymbol{Z})$ or $(X_2, \boldsymbol{Y}, \boldsymbol{Z})$. With these redefinitions of the Jacobian we can apply the following control function.

$$C(t) := \begin{cases} s \cdot \mathcal{J}^{\dagger}(S^v(t)) \cdot (Q^* - Q(t)) & : \|Q^* - Q(t)\| > \eta \\ 0 & : \quad else \end{cases} \tag{7}$$

A proportional control law is defined, i.e. the change is proportional to the deviation between desired and current measurements. Servoing factor s influences the step-size of approaching the goal place. The hand position is changed by a non-null vector $C(t)$ if desired and current positions in the image deviate more than a threshold η. According to our strategy, first the hand tip is servoed to the intersection point P_1 of the unknown optical axis with plane $(X_1, \boldsymbol{Y}, \boldsymbol{Z})$, and second to the intersection point P_2 with plane $(X_2, \boldsymbol{Y}, \boldsymbol{Z})$. Figure 4 shows for the hand servoing on one plane the succession of the hand tip extracted in the image, and the final point is located at the image center. The two resulting positions define a straight line in the manipulator coordinate system which is located near to the optical axis. The estimated line will be used to approach an object towards the camera for detailed inspection.

5 Servoing for shape inspection

Prior to the inspection phase of the peg-in-hole application the object must be grasped with the parallel jaw fingers of the robot hand [7, pp. 127-129, 210-

217]. The grasped object is carried to a specific pose in the viewing space of one camera. Concretely, the specific position is the intersection point of the optical axis and the bottom rectangle of the pyramid viewing space, and the specific orientation of the fingers is orthogonal to the optical axis. As an example, the orientation of a grasped cyclinder is such that the camera has an orthogonal view from the top or bottom, circular cyclinder face. Due to the large distance from the camera (most distant viewing position), the depiction of the circular face is small. In order to inspect the shape of an object face it is desirable to have the face depicted in the image as large as possible. For this purpose, the robot hand must be servoed along the optical axis towards the camera, which is illustrated in Figure 1 for one step of movement.

For this servoing process it is convenient to take as image measurements the appearance of the robot fingers. Due to their well-known shape the fingers can be extracted much easier (e.g. through Hough transformation) than the unknown shape of the object. We take the width of a robot finger (number of pixels) for defining current and desired measurement scalars (instead of vectors) $Q(t)$ and Q^*. Just as the measurements, also the control vector $C(t)$ is a scalar. With this definitions the control function of equation (7) can be applied for reaching the optimal viewing distance. The Jacobian may be simply defined by constant value 1, because servoing factor s can be used anyway for affecting the step-size. After having finished the approaching process we obtain an *acceptable size of the depicted object*, like in the first image of Figure 5.

Fig. 5. (a) Appropriate size of depicted situation; (b) Vertical appearance of robot fingers; (c) Binary image with object and fingers; (d) Extraction of grasped object.

The inspection of the object shape is based on extracting the relevant region in the image. Especially, the regions of the robot fingers must be suppressed. This task can be simplified by first applying once again hand servoing. It is intended to obtain a standardized (i.e. vertical) appearance of the robot fingers, as shown in the second image of Figure 5. For this purpose, the robot hand must rotate around the optical axis with the tip of the robot hand taken as the rotation center. For the servoing process we take as image measurement the finger tilt relative to the vertical image axis. Just as the measurements, also the control vector $C(t)$ is scalar. Therefore, a simple control procedure can be applied.

The usefulness of the *standardized finger appearance* is to be able to apply simple pattern matching techniques. We use the second image of Figure 5, then move the robot hand outside the viewing space and take an image from the background. The subtraction of both images reveals a binary image containing only the fingers and the object (third image of Figure 5). The suppression of the finger regions is reached with given finger patterns which were acquired in

an offline phase under similar viewing conditions. Actually, it is this matching process which can be performed efficiently due to the standardized finger appearance. The right image in Figure 5 is obtained which contains just the relevant object region. Undesired noisy effects (isolated white pixels) can be suppressed by applying simple morphological operations.

Our approach for describing the shape of the region is based on the autoregressive model proposed by Dubois [1]. It results in a characterizing vector of features which is invariant under region translation and rotation. The same approach is applied as well for describing the holes of the board which results in a characterizing vector for each hole, respectively. Based on euclidean metric one determines the hole whose shape is most similar to the shape of the peg. This concludes the inspection phase of the peg-in-hole application. The second phase consists in approaching the peg appropriately to the relevant hole.

6 Servoing for object assembly

The two cameras take images continually for the visual feedback control of approaching an object to a goal place. In each stereo image both the object and the goal place must be visible for determining the distance between current and desired measurement vectors, respectively. The critical issue is to extract the relevant features from the stereo images. For example, let us assume a *cylindrical object and a circular goal place* as shown in the first image of Figure 6.

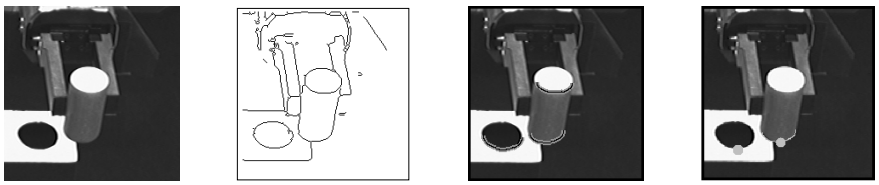

Fig. 6. (a) Cylindrical object, circular goal place; (b) Thresholded gradient magnitudes; (c) Extracted half ellipses; (d) Specific point on half ellipses of object and goal place.

The binarization based on thresholding the gradient magnitudes is shown in the second image of Figure 6. In the next step, a specific type of Hough transformation is applied for approximating and extracting half ellipses (third image in Figure 6). This specific shape is expected to occur at the goal place and at the top and bottom faces of the object. Instead of full ellipses we prefer half ellipses, concretely the lower part of full ellipses, because due to the specific camera arrangement this feature is visible throughout the complete process. From the bottom face of the object only the specific half ellipse is visible. The process of approaching the object to the goal place is organized such that the lower part of the goal ellipse remains visible, but the upper part may become occluded more and more by the object. The distance measurement between object and goal place just takes the half ellipse of the goal place and that from the bottom face of the object into account. For computing a kind of distance between the two relevant half ellipses we extract from each a specific point and based on this we can take any metric between 2D positions as distance

measurement. The fourth image in Figure 6 shows these two points, indicated by gray disks, on the object and the goal place.

The critical aspect of extracting points from a stereo pair of images is that *reasonable correspondences* must exist. A point of the first stereo image is in correspondence with a point of the second stereo image, if both originate from the same 3D point. In our application, the half ellipses extracted from the stereo images are the basis for determining corresponding points. However, this is by no means a trivial task, because the middle point of the contour of the half ellipse is not appropriate. The left picture of Figure 7 can be used for explanation. A virtual scene consists of a circle which is contained in a square (top part of left picture). Each of the two cameras produces a specific image, in which an ellipse is contained in a quadrangle (bottom part of left picture). The two dotted curves near the circle indicate that different parts of the circle are depicted as lower part of the ellipse in each image. Consequently, the middle points p_1 and p_2 on the lower part of the two ellipses originate from different points P_1 and P_2 in the scene, i.e. points p_1 and p_2 do not correspond. Instead, the right picture of Figure 7 illustrates an approach for determining corresponding points.

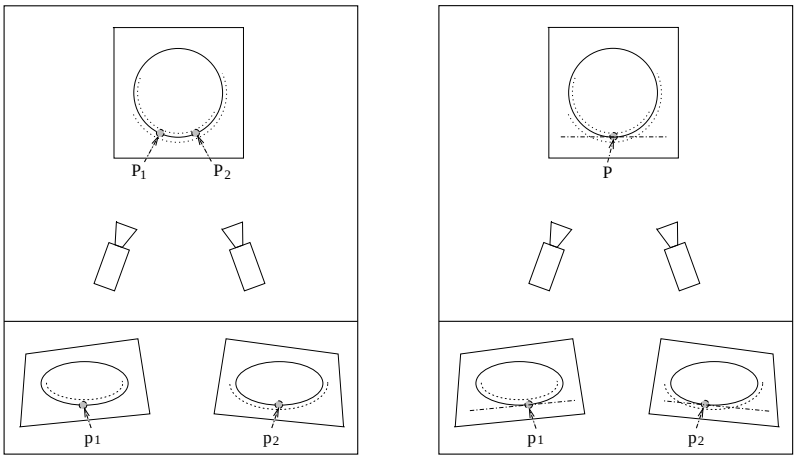

Fig. 7. (a) Extracted image points p_1, p_2 originate from different scene points P_1, P_2; (b) Extracted image points correspond, i.e. originate from one scene point P.

We make use of a specific geometric relation which is invariant under geometric projection. Virtually, the bottom line of the square is translated to the circle which results in the tangent point P. This procedure is done as well in both stereo images, i.e. translating the bottom line of the quadrangle parallel towards the ellipse to reach the tangent points p_1 and p_2. Due to different perspectives the two bottom lines have different orientations and therefore the resulting tangent points are different from those extracted previously (compare bottom parts in left and right picture of Figure 7). It is observed easily that the new tangent points p_1 and p_2 correspond, i.e. originate from the same scene point P. This kind of projective compatibility can be exploited for our peg-in-hole application. The boundary of the board, which contains the holes, can be used as supporting

context for stereo matching. Accordingly, both the board and the object must be fully included in the viewing space of both stereo cameras, respectively. For each stereo image the orientation of the bottom boundary line can be used for determining relevant tangent points at the relevant ellipse, i.e. virtually move the lines to the ellipses and keep orientation. Tangent points must be extracted at the half ellipse of the goal place and at the half ellipse of the bottom face of the object. These points have already been shown in the fourth image of Figure 6.

For defining the control vector we must describe the relationship between displacements of the robot hand and the resulting displacements in the two stereo images taken by the stereo cameras. For this purpose we introduce two Jacobians $\mathcal{J}_1(P)$ and $\mathcal{J}_2(P)$ which depend on the current position P of the hand tip. If we would multiply the Jacobian $\mathcal{J}_1(P)$ (respectively Jacobian $\mathcal{J}_2(P)$) with a displacement vector of the hand position, then the product would reveal the displacement vector in the left image (respectively in the right image). The two Jacobians are joined together which results in a (4×3) matrix depending on P.

$$\mathcal{J}(P) := \begin{pmatrix} \mathcal{J}_1(P) \\ \mathcal{J}_2(P) \end{pmatrix} \tag{8}$$

In order to transform a desired change from stereo image coordinates into manipulator coordinates the pseudo inverse $J^{\dagger}(P)$ is computed.

$$\mathcal{J}^{\dagger}(P) := \left(\mathcal{J}^T(P) \cdot \mathcal{J}(P)\right)^{-1} \cdot \mathcal{J}^T(P) \tag{9}$$

The current position $P(t)$ of the hand tip defines the variable state vector $S^v(t)$. The desired measurement vector Q^* is a 4D vector comprising the 2D positions of a certain point of the goal place in the stereo images. The current measurement vector $Q(t)$ represents the stereo 2D positions of a relevant point on the object.

$$Q^* := \begin{pmatrix} p_1^* \\ p_2^* \end{pmatrix}; \quad Q(t) := \begin{pmatrix} p_1(t) \\ p_2(t) \end{pmatrix} \tag{10}$$

With these new definitions we can apply control function of equation (7).

7 Peg-in-hole application for other objects

The basic assumption behind the presented technique is that the peg can be inserted successfully by taking only the shape of the bottom object face into account. Accordingly, the object surface must be composed of a top and a bottom face, which are parallel and of equal shape, and the other faces must be orthogonal to them. Apart from cylinders this constraint also holds for cuboids, whose treatment will be mentioned briefly (Figure 8). The procedures involved in the inspection phase can be applied without any change. However, in the insertion phase we must consider that the object is not rotation-symmetric. In addition to the positions of hole and object, also the orientations have to be taken into account. Hough transformation and strategies for line organization are applied for extracting the boundaries of object and hole, respectively [7, pp. 29-98]. Based on the hole boundary and the top face boundary of the object we

determine hole and object orientation. Furthermore, we take the middle point of two appropriately selected boundary lines of hole and object to determine their positions. Altogether, the current measurement vector $Q(t)$ consists of 6 components with 3 for each stereo image. These are composed of one scalar for the orientation and 2 scalars for the position of the object. Similary, the desired measurement vector Q^* is defined for the hole. The control vector $C(t)$ consists of 4 components, i.e. three for the position and one for the horizontal orientation of the robot hand. Based on these definitions we determine the Jacobian and apply the control function of equation (7). Figure 8 shows the peg-in-hole application for the cuboid object, which includes in the second image the object boundary and the selected point for defining the measurement vector.

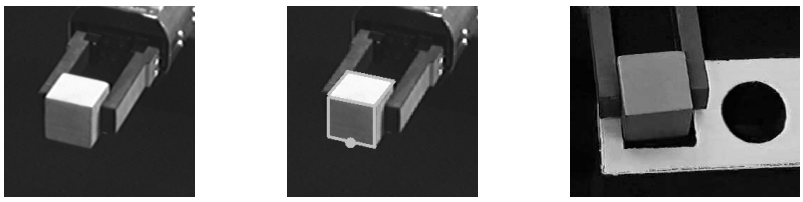

Fig. 8. (a) Grasped cuboid object; (b) Set of object boundary lines, selected point specifying object position in the image; (c) Insertion of cuboid into rectangular hole.

8 Summary

For peg-in-hole applications we used a two-component robot system which consists of a robot manipulator (including a parallel jaw gripper) and a robot head (including monochrome stereo cameras). The usefulness of image-based hand-effector servoing was demonstrated for characterizing the camera-manipulator relation, for optimal viewing and inspecting the object, and for appropriately approaching the object to the relevant hole. In our current implementation, one servoing cycle for inserting the cylindrical peg requires about 0.5 seconds. Generally, the velocity depends on the complexity of the object shape.

References

1. Dubois, S., Glanz, F.: An autoregressive model approach to two-dimensional shape classification. IEEE Trans. on Patt. Anal. and Mach. Intel. **8** (1986) 55-66
2. Faugeras, O.: Three-Dimensional Computer Vision. The MIT Press (1993)
3. Hashimoto, K.: Visual Servoing. World Scientific Publishing (1993)
4. Horaud, R., Chaumette, F. (eds.): International Journal of Computer Vision, special issue on Image-based Robot Servoing **37** (2000)
5. Hutchinson, S., Hager, G., Corke, P.: A tutorial on visual servo control. IEEE Trans. on Robotics and Automation **12** (1996) 651-670
6. Lanzetta, M., Dini, G.: An integrated vision-force system for peg-in-hole assembly operations. Intel. Comp. in Manufact. Eng. (1998) 615-621
7. Pauli, J.: Development of camera-equipped robot systems. Christian-Albrechts-Universität zu Kiel, Institut für Informatik, Technical Report **9904** (2000)