Vision-based integrated system for object inspection and handling

Josef Pauli *,[a] , Arne Schmidt [b] , Gerald Sommer [a]

[a] *Christian-Albrechts-Universität zu Kiel, Institut für Informatik und Praktische Mathematik, Preußerstraße 1–9, D-24105 Kiel, Germany*

[b] *Heidelberger Druckmaschinen AG, Center of Expertise Engineering, Dr.-Hell-Straße, D-24107 Kiel, Germany*

Abstract

Image-based effector servoing is a process of perception-action cycles for handling a robot effector under continual visual feedback. This paper applies visual servoing mechanisms not only for handling objects, but also for camera calibration and object inspection. A 6-DOF manipulator and a stereo camera head are mounted on separate platforms and are steered independently. In a first phase (calibration phase), camera features are determined like the optical axes and the fields of sharp view. In the second 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 the object shape is inspected in detail. In the third phase (assembly phase), the system localizes a board containing holes of different shapes, determines the hole which fits most appropriate to the object shape, then approaches and arranges the object appropriately. The final object insertion is based on haptic sensors, but is not treated in the paper. At present, the robot system has the competence to handle cylindrical and cuboid pegs. For handling other object categories the system can be extended with more sophisticated strategies of the inspection and/or assembly phase.

Key words: Visual feedback control; Optical axis estimation; Shape inspection; Object assembly.

* Corresponding author. Tel. +49-431-560484; Fax. +4 9-431-560481.
Email address: jpa@ks.informatik.uni-kiel.de (Josef Pauli).
URL: www.ks.informatik.uni-kiel.de/∼jpa/ (Josef Pauli).

Article published in: **Robotics and Autonomous Systems 37 (2001) 297-309**

1 Introduction

Image-based robot servoing (short, *visual servoing*) is the backbone of Robot Vision systems. The book edited by Hashimoto [8] 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. [10]. Quite recently, a special issue of the International Journal on Computer Vision has been devoted to image-based robot servoing [9].

Frequently, papers on visual servoing treat isolated sub-tasks, e.g. approaching an object to a target location [7]. Opposed to that, this work demonstrates exemplary the usefulness of servoing for treating a spectrum of 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 assembly*. Furthermore, minimalism principles are considered by extracting just the necessary image information and *avoiding 3D reconstruction*, which leads to real-time usage. Related to the application of *peg-in-hole assembly operations* it is favourable to integrate video and force information [11]. However, this paper focuses on the vision-related sub-tasks of the overall peg-in-hole application which take place primarily in the run-up phase prior to the actual insertion phase.

As an overview, the paper describes the components of the robot system (Section 2), presents the general measurement-based control procedure (Section 3), exploits the manipulator agility for estimating projection matrices (Section 4), and applies servoing mechanisms for determining the optical camera axis (Section 5). Then, the work presents an approach for robotic object grasping (Section 6), applies servoing mechanisms for optimally viewing and inspecting the object (Section 7, see also Fig. 1), and applies servoing mechanisms to suitably approach the object to the relevant hole (Sections 8 and 9, see also Fig. 2). Finally, the performance of the proposed vision-based integrated system is measured by the accuracy of object positioning (Section 10). Various experiments are presented for demonstrating the usefulness of visual servoing even in cases of unintentional disturbance of the camera-robot relation.

Industrial systems often work in two consecutive phases consisting of *gross motion* and *fine motion* [12, pp. 452-453]. This system attains high positioning accuracies (i.e. mean deviations of $1mm$ to $3mm$ from desired positions) and therefore is highly appreciated in the *gross motion phase*. For example, in a car dismantling application [2] the robot arm carries a screw driver near to the screws of a wheel (i.e. aligned with the turning axes of the screws) and close-range force/torque sensors are used for the unscrew process. In certain applications like medical robotics the gross motion phase may be done automatically and the fine motion phase manually by a person [5], e.g. automatically approaching a surgical instrument towards the operating area and interactively performing the surgical operation by the robot-supported handling of the instrument in a master/slave fashion. In medical and other applications the purpose of fine motion may also be a close-range microscopic imaging. Finally, the presented system can also be used in industrial applications which only need the *gross motion* (but omit fine motion), e.g. in an autonomous sorting system an air ejection effector or a magnetic effector can be positioned approximately at the right place in order to remove the relevant objects [4].

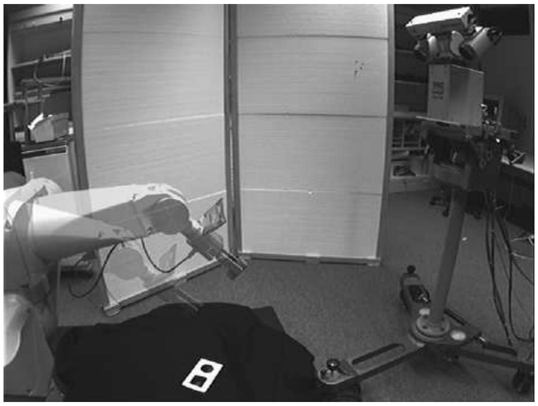

Fig. 1. Robot head and manipulator, approaching an object towards a camera for shape inspection.

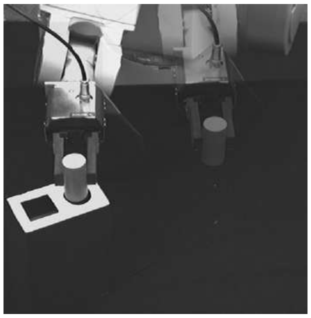

Fig. 2. Vision-based approaching a cylindrical peg towards a circular hole.

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 into arbitrary positions and orientations 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 platform and is equipped with motorized pan, tilt, and vergence degrees-of-freedom (DOF). Additionally, the stereo camera has motorized zooming and focusing facilities.

3 Procedure of measurement-based control

The robot system is characterized by a *fixed state vector* S^c (e.g. lengths of the links) which is inherent constant in the system, and by a *variable state vector* $S^v(t)$ (e.g. angles of the joints) which can be changed through a *vector of control signals* $C(t)$ at time t. Both the 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}, which is in the simplest case a vector 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 used 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} must describe 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.

$$\mathcal{M} := \begin{pmatrix} m_1^v \\ m_2^v \\ m_3^v \end{pmatrix} \tag{4}$$

$$\begin{aligned} m_1^v &:= (m_{11}, m_{12}, m_{13}, m_{14}) \\ m_2^v &:= (m_{21}, m_{22}, m_{23}, m_{24}) \\ m_3^v &:= (m_{31}, m_{32}, m_{33}, m_{34}) \end{aligned} \tag{5}$$

The usage of the projection matrix is specified within the following context [6, 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 as follows.

$$p := f^{pr}(P) = \frac{1}{\xi} \cdot \mathcal{M} \cdot P \tag{6}$$

$$f^{pr}(P) := \begin{pmatrix} f_1^{pr}(P) \\ f_2^{pr}(P) \\ f_3^{pr}(P) \end{pmatrix} \tag{7}$$

$$\xi := m_3^v \cdot P \tag{8}$$

The scalar parameters m_{ij} of matrix $\mathcal{M}$ represent a combination of extrinsic and intrinsic camera parameters, respectively. The specific definition of the normalizing factor ξ in equation (8) guarantees that function $f_3^{pr}(P)$ is constant 1, i.e. the homogeneous image coordinates of position p are given in normalized form. In contrast to other approaches (e.g. [7]), the extrinsic parameters, like the camera orientation, or intrinsic parameters, like the focal length, are never computed or used explicitly. According to the minimalism principles (see Section 1), the camera parameters are left implicit in the elements of matrix $\mathcal{M}$, which simplifies the computation of the *Jacobian* $\mathcal{J}$. For function f^{pr} in equation (6) the Jacobian is computed as follows.

$$\mathcal{J}(P) := \begin{pmatrix} \frac{\partial f_1^{pr}}{\partial X}(P) & \frac{\partial f_1^{pr}}{\partial Y}(P) & \frac{\partial f_1^{pr}}{\partial Z}(P) \\ \frac{\partial f_2^{pr}}{\partial X}(P) & \frac{\partial f_2^{pr}}{\partial Y}(P) & \frac{\partial f_2^{pr}}{\partial Z}(P) \end{pmatrix}$$

$$:= \begin{pmatrix} \frac{m_{11} \cdot m_3^v \cdot P - m_{31} \cdot m_1^v \cdot P}{(m_3^v \cdot P) \cdot (m_3^v \cdot P)} & \frac{m_{12} \cdot m_3^v \cdot P - m_{32} \cdot m_1^v \cdot P}{(m_3^v \cdot P) \cdot (m_3^v \cdot P)} & \frac{m_{13} \cdot m_3^v \cdot P - m_{33} \cdot m_1^v \cdot P}{(m_3^v \cdot P) \cdot (m_3^v \cdot P)} \\ \\ \frac{m_{21} \cdot m_3^v \cdot P - m_{31} \cdot m_2^v \cdot P}{(m_3^v \cdot P) \cdot (m_3^v \cdot P)} & \frac{m_{22} \cdot m_3^v \cdot P - m_{32} \cdot m_2^v \cdot P}{(m_3^v \cdot P) \cdot (m_3^v \cdot P)} & \frac{m_{23} \cdot m_3^v \cdot P - m_{33} \cdot m_2^v \cdot P}{(m_3^v \cdot P) \cdot (m_3^v \cdot P)} \end{pmatrix} \tag{9}$$

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 needed (see following sections).

4 Estimating the projection matrices

For the robot system, consisting of arm and bisight head, two projection matrices $\mathcal{M}_1$ and $\mathcal{M}_2$ have to be estimated, as introduced generally in equations (4) and (5), which are related to the two stereo cameras. Based on training samples of corresponding 3D and 2D coordinate vectors the matrices are determined by simple linear methods, e.g. singular value decomposition [14, pp. 59-70]. For a certain real-world point, the 3D coordinate vector is specified in the basis coordinate system of the robot arm, and the corresponding 2D coordinate vectors are specified in the two image coordinate systems of the stereo cameras (on the robot head), respectively.

In order to obtain the training samples of corresponding coordinate vectors, full advantage of the agility of the robot arm can be taken. The robot hand moves in the working space systematically, stops on equidistant places, and 3D positions of the robot hand are carefully recorded. These 3D vectors are supplied by the control unit of the robot arm. Additionally, at each stopping place the hand tip of the robot arm must be localized in both stereo images. For detecting the hand tip, the boundaries of the robot fingers are extracted by Hough transformation, the virtual hand axis and the end straight line are determined, and the intersection point (middle point between the two finger tips) is taken as hand tip (see Fig. 3). For increasing the reliability of detection, this boundary-based location is verified and/or adjusted by normalized cross correlation, i.e. matching based on a template of the finger tips. [1]

The strategy of using the robot arm itself for determining the arm–head relation is advantageous in several aspects. First, an artificial calibration object is not needed. Second, training samples can be taken both from the surface and within the working space. Third, the number of samples for approximating the function is variable due to steerable distances between the stopping

[1] Quite recently, Baek et al. presented an alternative approach of end-effector tracking [1].

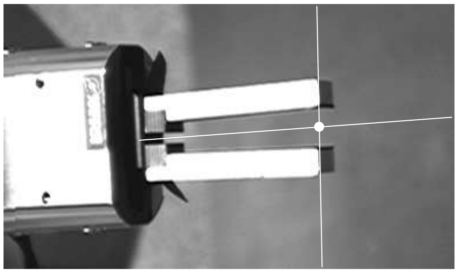

Fig. 3. Extraction of virtual hand axis, end straight line, and intersection point as hand tip.

places. Fourth, the arm–head relation is computed relative to the basis coordinate system of the robot arm directly, which is the relevant coordinate system for steering the robot hand.

5 Servoing for estimating the optical axis

For estimating the *optical axis of a camera* relative to the basis coordinate system of the manipulator, image-based hand-effector servoing is applied. 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, a 3D position can be obtained 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 $(\vec{Y}, \vec{Z})$ plane with constant offsets X_1 and X_2 on the $\vec{X}$-axis. The movement of the hand-effector is restricted just to these planes (see Fig. 4). Accordingly, the generic definition of the Jacobian $\mathcal{J}$ in equation (9) can be restricted to the second and third columns, because the coordinates on the $\vec{X}$-axis are fixed. A quadratic Jacobian matrix is obtained (with two rows and columns) which must be inverted, i.e. $\mathcal{J}^{\dagger}(P) := \mathcal{J}^{-1}(P)$.

The desired measurement vector Q^* is defined as the image center point and the current measurement vector $Q(t)$ as the 2D image location of the hand tip. The variable state vector $S^v(t)$ consists of the two variable coordinates of the tool center point in the selected plane $(X_1, \vec{Y}, \vec{Z})$ or $(X_2, \vec{Y}, \vec{Z})$. With these redefinitions of the Jacobian the following control function can be applied.

$$C(t) := \begin{cases} s \cdot \mathcal{J}^{\dagger}(S^v(t)) \cdot (Q^* - Q(t)) & : \|Q^* - Q(t)\| > \eta \\ 0 & : \quad else \end{cases} \tag{10}$$

Servoing factor s is used to influence 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 this strategy, first the hand tip is servoed to the intersection point P_1 of the unknown optical axis with the plane $(X_1, \vec{Y}, \vec{Z})$, and second to the intersection point P_2 with plane $(X_2, \vec{Y}, \vec{Z})$. For locating the hand tip in the images and thus obtaining the current measurement vector $Q(t)$, the detection approach mentioned in Section 4 is used. Fig. 5 shows for the hand-effector servoing on one plane the succession of the hand tip extracted in the image, and the final point is located at the image center.

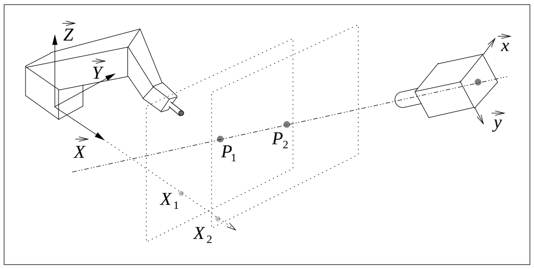

Fig. 4. Hand-effector servoing for estimating the optical axis of a camera.

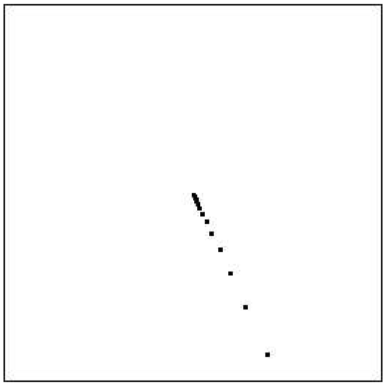

Fig. 5. Course of detected hand tip towards image center.

The two resulting 3D positions define a straight line in the manipulator coordinate system which is located near to the optical axis. For the purpose of detailed inspection an object can be moved towards the camera along this estimated line.

Image-based hand-effector servoing is also a means for constructing the *field of sharp view* of the camera, which can be approximated as a truncated pyramid with top and bottom rectangles normal to the estimated optical axis (for details see [13, pp. 248-252]).

6 Strategy for object grasping

Prior to the inspection phase the object must be grasped with the parallel jaw fingers of the robot hand. A rather simple approach will be mentioned which works successful for a large spectrum of different objects. The grasping is performed at an a priori specified height (Z-coordinate) with the fingers horizontal to the ground plane. Therefore, the only unknowns are the 2D grasping position (X- and Y-coordinates) and the horizontal grasping (finger) orientation. For this purpose, one color camera with the optical axis directed vertical (approximately) to the ground plane is used. The color camera is in addition to the previously mentioned stereo cameras. Based on YUV color space just the U- and V-components are used which are independent from luminance. Assuming homogenous background it is easy to extract an image region for each object, apply morphological filtering for smoothing the boundary, extract the contour and approximate a polygon, respectively.

Fig. 6 shows a set of objects (including the cylindrical and cuboid objects) which can be grasped successfully. Fig. 7 shows the binary image of the polygonal object boundaries which are relevant for grasping. The longest straight line of each polygon is assumed to indicate the dominant orientation of the object. The orientation of this dominant line is taken as the finger orientation and the center of gravity of the object region as grasping position. The transformation from image to manipulator coordinates is based on a simple camera calibration. Alternatively to this boundary-based approach, also appearance-based approaches for recognizing grasping situations [13, pp. 131-134] may be applied.

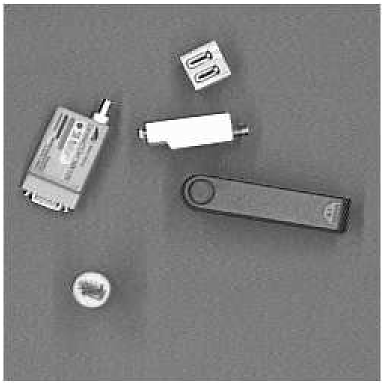

Fig. 6. Arrangement of objects which can be grasped consecutively.

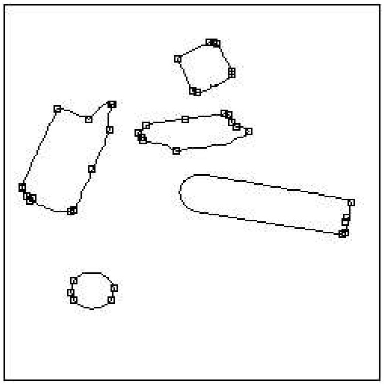

Fig. 7. Extracted object boundaries for determining grasping poses.

7 Servoing for shape inspection

Based on the grasping competence the robot hand carries the grasped object 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 Fig. 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. The thickness of a robot finger (number of pixels) is taken for defining current and desired measurement scalars $Q(t)$ and Q^* (special case of vectors). Just as the measurement, also the control vector $C(t)$ is a scalar. With this definitions the control function of equation (10) 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, an *acceptable size of the depicted object* is obtained, like in Fig. 8(a). 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-effector servoing. It is intended to obtain a standardized (i.e. vertical) appearance of the robot fingers, as shown in Fig. 8(b). 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 the finger tilt relative to the vertical image axis is taken. Just as the measurement, also the control vector $C(t)$ is scalar, and 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. The image of Fig. 8(b) is used as basis, then the robot hand is moved outside the viewing space and an image from the background is taken. The subtraction of both images reveals a binary image containing only the fingers and the object (image of Fig. 8(c)). The suppression of the finger regions is obtained 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 image in Fig. 8(d) is obtained which contains just the relevant object region. Undesired noisy effects (isolated white pixels) can be suppressed by applying simple morphological operations.

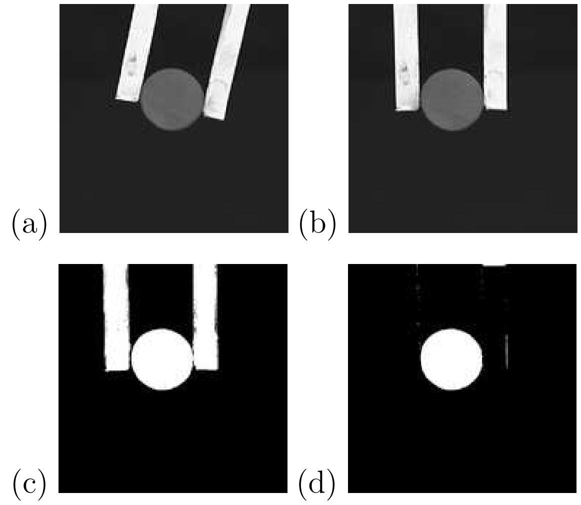

Fig. 8. (a) Appropriate size of depicted grasping situation; (b) Vertical appearance of the robot fingers; (c) Binary image consisting of grasped object and robot fingers; (d) Extraction of grasped object.

The approach for describing the shape of the region is based on the autoregressive model proposed by Dubois [3]. 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 next phase consists in approaching and arranging the peg appropriately to the relevant hole.

8 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, a *cylindrical object* and a *circular goal place* are assumed as shown in the image of Fig. 9(a).

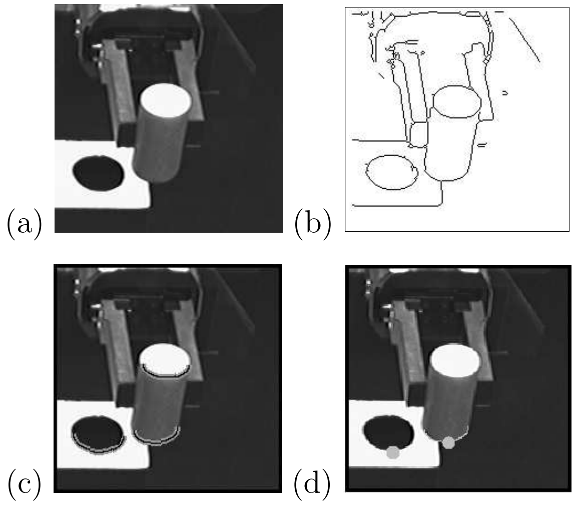

Fig. 9. (a) Cylindrical object and circular goal place; (b) Binary image of thresholded gradient magnitudes; (c) Extracted half ellipses; (d) Specific point on half ellipses of object and goal place.

The binarization is based on thresholding the gradient magnitudes, as shown in the image of Fig. 9(b). In the next step, a specific type of Hough transformation is applied for approximating and extracting half ellipses (image in Fig. 9(c)). 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, half ellipses are preferred, 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 the distance between the locations of the two relevant half ellipses, from each a specific point is extracted and based on this any metric between 2D positions can be taken as distance measurement. The image in Fig. 9(d) shows these two points, indicated by gray disks, on the object and the goal place. In the following, the approach for determining these two points is explained in detail.

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 this 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 picture (a) of Fig. 10 can be used for explanation.

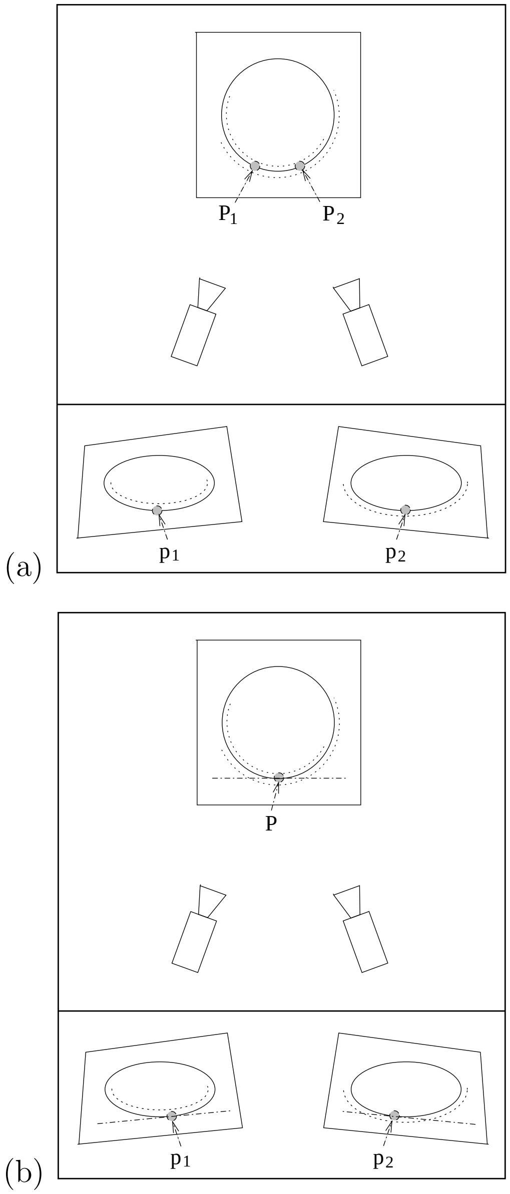

Fig. 10. (a) Extracted image points p_1 and p_2 originate from different scene points P_1 and P_2; (b) Extracted image points are corresponding, i.e. originate from one scene point P.

A virtual scene consists of a circle which is contained in a square (top part of picture (a)). Each of the two cameras produces a specific image, in which an ellipse is contained in a quadrangle (bottom part of picture (a)). 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. In consequence of this, 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 picture (b) of Fig. 10 illustrates an approach for determining corresponding points. A specific geometric relation isused 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 pictures (a) and (b) of Fig. 10). 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 the peg-in-hole assemply application.

The boundary of the board, which contains the holes, can be used as supporting context for stereo matching (see image in Fig. 11(a)). Accordingly, both the board and the object must be fully included (or at least significant parts) in the viewing space of both stereo cameras, respectively (see image in Fig. 9(a)). For each stereo image the orientation of the bottom boundary line can be used for determining relevant tangent points at the relevant ellipse. Virtually, the line is moved to the ellipses while keeping orientation. Tangent points must be extracted at the half ellipse of the goal place (see image in Fig. 11(b)) and at the half ellipse of the bottom face of the object (see image in Fig. 11(c)). These points have already been shown in the fourth image of Fig. 9.

For defining the control vector, the relationship between displacements of the robot hand and the resulting displacements in the two stereo images (taken by the stereo cameras) must be described. For this purpose two Jacobians $\mathcal{J}_1(P)$ and $\mathcal{J}_2(P)$ are introduced which depend on the current position P of the hand tip. The Jacobians are computed for the two projection matrices $\mathcal{M}_1$ and $\mathcal{M}_2$ introduced in Section 4. If the Jacobian $\mathcal{J}_1(P)$ (respectively Jacobian $\mathcal{J}_2(P)$) would be multiplied 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 simply joined together which results in a (4×3) matrix depending on P.

$$\mathcal{J}(P) := \begin{pmatrix} \mathcal{J}_1(P) \\ \mathcal{J}_2(P) \end{pmatrix} \tag{11}$$

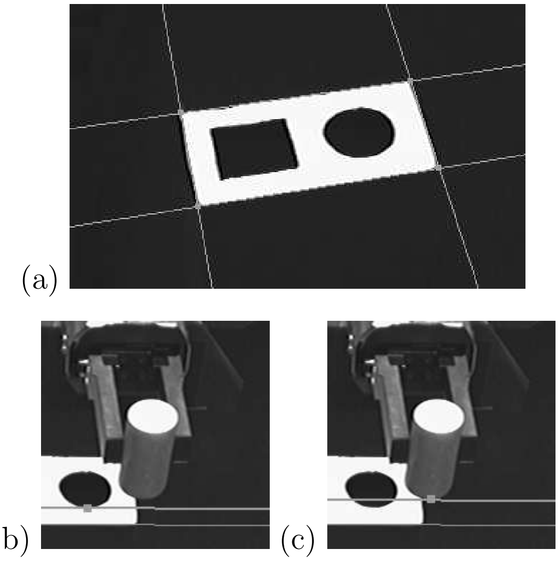

Fig. 11. (a) Boundary in the image of the board; (b) Determining the goal position at the image ellipse of the circular hole; (c) Determining the current position at the image ellipse of the bottom face of the cylindrical object.

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

$$\mathcal{J}^{\dagger}(P) := \left(\mathcal{J}^{T}(P) \cdot \mathcal{J}(P)\right)^{-1} \cdot \mathcal{J}^{T}(P) \tag{12}$$

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 (see above).

$$Q^{*} := \begin{pmatrix} p_1^{*} \\ p_2^{*} \end{pmatrix}; \quad Q(t) := \begin{pmatrix} p_1(t) \\ p_2(t) \end{pmatrix} \tag{13}$$

With these new definitions, i.e. $J^{\dagger}(P)$, $S^{v}(t)$, $Q(t)$, and Q^{*}, the control function of equation (10) can be applied. Actually, a proportional control law (P-controller) is defined, meaning that the change is proportional to the deviation between the desired and the current position. [2]

[2] Alternatively, the P-controller can be combined with an integral and a derivative control law to construct a PID-controller. However, the P-controller is good enough for this simple control task.

9 Handling other categories of object shapes

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 exemplary, whose treatment will be mentioned briefly. The procedures involved in the inspection phase can be applied without any change. However, in the assembly phase it must be 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 [13, pp. 25-99]. Based on the hole boundary and the top face boundary of the object we determine hole and object orientation. Furthermore, the middle point of two appropriately selected boundary lines of hole and object is taken 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 the Jacobian is determined and the control function of equation (10) is applied. Fig. 12 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.

10 Experiments

Strategies of repetitive perception-action cycles are favourable in comparison with ultimate look-and-move strategies. In this work, the hand-effector is gradually moved under continual visual feedback. The system has the chance of reacting appropriately and changing a prior behavior in case of an unforeseen environmental situation. For example, the process of object assembly can be changed if the board for object insertion has been moved. In this section, experiments are carried out concerning a second type of imponderables, i.e. unintentional disturbance of the robot head. Image-based hand-effector servoing is based on the Jacobians of two projection matrices which are determined for the stereo cameras of the robot head. The disturbance of the robot head (e.g. change of pan or tilt) causes inaccurate projection matrices, which perhaps makes a re-calibration necessary. However, it is known that servoing mechanisms have the potential to *balance out inaccurate approximations*

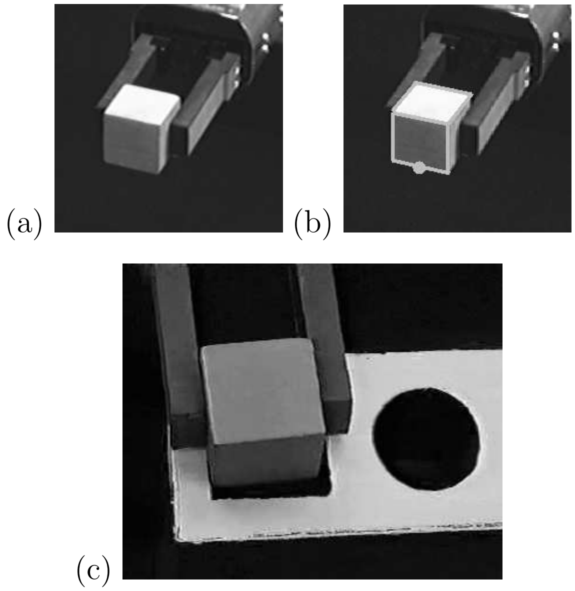

Fig. 12. (a) Grasped cuboid object; (b) Set of object boundary lines and a selected point which specifies the object position in the image; (c) Insertion of the cuboid into the rectangular hole.

and thus avoid a re-calibration. The purpose of the following experiments is to concretely determine this potential for the arrangement of robot arm and head.

Throughout all experiments the visual servoing task is to move the robot hand to a certain *real-world point (goal place)* starting at an arbitrary point in the working space. For the robot hand a cuboid working space is assumed with the extension $200mm \times 400mm \times 200mm$, the distance of the cuboid center to the robot head is $1400mm$, the optical axes of the stereo cameras are directed to the cuboid center and the focal lengths are tuned equally to $17mm$. The generic control function of equation (10) is used with the Jacobian as defined in equation (11). For simplifying goal detection in the images (i.e. determine Q^*) a light spot is put to the goal place, and for the detection of the current hand position (i.e. determine $Q(t)$) the approach mentioned in Section 4 is applied. The center of the working space is taken as the goal place, whose 3D coordinates are known in the coordinate system of the robot arm. Additionally, the 3D position of the robot hand, which is reached after a certain number of servoing cycles, can be obtained from the control unit. Therefore, the distance of the robot hand from the desired goal place can be taken as a criterion for evaluating the servoing mechanism.

In the first experiment, *different versions of projection matrices* are used, which are obtained by different numbers of training samples. The smaller the training set the less accurate the projection matrices, however, the larger the training set the more costly the calibration process. It is desirable to spend minimal effort in the calibration phase and obtain an acceptable servoing behavior in the application phase. By applying the approach of Section 4, three sets are acquired consisting of 225, 45, and 8 equally distributed training samples, respectively. In particular, the latter set consists just of the 8 corner points of the cuboid working space. The servoing factor is set to $s := 0.5$, and 11 servoing cycles have been performed. Table 1 shows the final distance between desired and actual hand position for the three cases, respectively, which means the error of hand positioning (or more generally of object positioning). Only small improvements are obtained when taking a large training set instead of small ones. The distances in the individual coordinates are between $0.09mm$ and $1.3mm$. A training set consisting of 45 elements is sufficient for the application and will also be used in the following experiments.

Table 1
Errors of robot hand positioning for alternative system calibrations, using different numbers of training samples.

Number of training samples	Positioning error ($\triangle X$, $\triangle Y$, $\triangle Z$) in mm
225	(0.70, 0.09, 0.40)
45	(0.10, 0.20, 0.03)
8	(1.30, 0.30, 0.50)

In the second experiment, *different values for servoing factor s* are taken in order to modify the servoing mechanism. Typically, there is a trade-off between the minimal number of servoing cycles and avoidance of oversteering, which can be treated by this factor. Servoing cycles are performed until the distance between desired and current position becomes less than $2mm$ in each coordinate. Table 2 shows for three values of the servoing factor the number of cycles, the final positioning errors, and the oversteering feature. A compromise is reached with factor $s := 0.7$ which avoids oversteering under a small number of 8 servoing cycles.

Table 2
Errors of robot hand positioning and oversteering statement for different values of the servoing factor.

Servoing factor s	Number of cycles	Positioning error ($\triangle X$, $\triangle Y$, $\triangle Z$) in mm	Oversteering of controller
0.50	11	(0.10, 0.20, 0.03)	no
0.70	8	(0.50, 0.70, 0.40)	no
1.00	3	(2.00, 0.10, 1.40)	yes

In the third experiment, *the robot head is disturbed* consciously in order to cause an erroneous calibration. Concretely, the pan and tilt degrees-of-freedom of the robot head and the focal lengths of the cameras are changed. Change of the pan angle is up to 25°, change of the tilt angle is up to 5°, change of focal length is up to $-6mm$. Servoing cycles are performed until the distance between desired and current position of the robot hand becomes less than $4mm$ in each coordinate, and the servoing factor is set to $s := 0.5$. Table 3 shows for various combinations of changes that the servoing mechanism is capable to balance out different kinds of disturbance of the robot head. For example, according to the fifth row the distance of the robot hand from the goal position (positioning error) is $2.00mm$, $0.08mm$, and $3.00mm$ in the X-, Y-, and Z-coordinate, respectively. This result is obtained after the execution of 20 servoing cycles. However, if an ultimate look-and-move strategy would be applied, the distance of the robot hand from the goal position would be $3mm$, $159mm$, and $91mm$ in the X-, Y-, and Z-coordinate, respectively. Altogether, this final experiment shows the potential usefulness of the servoing mechanisms.

Table 3
Errors of robot hand positioning and numbers of needed servoing cycles under different kinds of disturbance of the robot head.

Changes of head/camera			Number of	Positioning error
Pan angle	Tilt angle	Focal length	cycles	($\triangle X$, $\triangle Y$, $\triangle Z$) in mm
5°	0°	$0.0\ mm$	9	(2.50, 1.80, 2.80)
10°	0°	$0.0\ mm$	11	(0.70, 0.20, 1.00)
10°	5°	$0.0\ mm$	20	(0.80, 0.30, 1.30)
0°	0°	$-5.75\ mm$	20	(2.00, 0.10, 2.80)
10°	5°	$-5.75\ mm$	20	(2.00, 0.08, 3.00)
25°	5°	$-5.75\ mm$	22	(1.30, 0.50, 3.20)

The presented series of experiments demonstrate that the system has the potential to attain high positioning accuracies. A coarse camera/robot calibration (using only a small set of training elements) is preferred and serious unintentional changes of head/camera attributes are considered. Despite of that, the servoing mechanisms succeed to limit the final positioning error within an interval of $0mm$ to $3mm$. Various industrial applications are conceivable for using this system as mentioned in section 1. Specifically, for a peg-in-hole application the final insertion can be completed if the clearance between peg and hole is more than $3mm$ in the horizontal coordinates.

11 Summary and discussion

For object inspection and handling applications a two-component robot system has been used 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 three purposes: (a) characterizing the cameras and the camera-manipulator relation, (b) optimal viewing and inspecting the object, and (c) appropriately approaching and arranging the object to a goal place. By continual visual feedback the system compensates an unintentional disturbance of the camera-manipulator calibration, avoids 3D reconstuction by leaving the relevant information implicit in the correspondences between the stereo images, and additionally may take care for unforeseen changes in the scene.

The most serious problem is image-based situation recognition in real-time (e.g. video rate), which is the precondition to determine appropriate control signals. In the current implementation, one servoing cycle for arranging the cylindrical peg requires about 0.5 seconds. Generally, the velocity depends on the complexity of the object shape. Prior to the servoing cycles certain intermediate and goal situations must be arranged manually and from the images thereof a set of appropriate operators for image analysis must be acquired. As these operators are grounded in actual situations the application during the servoing cycle promises to be successful. For making recognition more easy and efficient it is important as well to exploit the possibility of active viewing. In the presented work this has been considered for inspecting the object shape.

The interested reader may wish to look a relevant video, which can be acquired as mpeg-file via the World-Wide-Web under the URL
http://www.ks.informatik.uni-kiel.de/∼jpa/put_peg.html
The video shows the assembly phase of a peg-in-hole application. The treatment of the insertion phase requires an incorporation of additional haptic/force information, which is deferred to future work.

References

[1] S. Baek, D.-S. Park, J. Cho, Y.-B. Lee, A robot endeffector tracking system based on feedforward neural networks, Robotics and Autonomous Systems 28 (1999), 43-52.

[2] U. Büker, S. Drüe, N. Götze, G. Hartmann, An active object recognition system for disassembly tasks, in H. Araujo, J. Dias, (eds.), Proceedings of the 7th International Symposium on Intelligent Robotic Systems, 1999, pp. 79-88.

[3] S. Dubois, F. Glanz, An autoregressive model approach to two-dimensional shape classification, IEEE Transactions on Pattern Analysis and Machine Intelligence 8 (1986), 55-66.

[4] Dunkley International, Inc., Machine Vision Systems for Quality Control, http://www.dunkleyintl.net/

[5] D. Engel, J. Raczkowsky, H. Wörn, An intuitive man-machine interface for robot control in craniofacial surgery, The IFAC Symposium on Human-Machine Systems, 2001.

[6] O. Faugeras, Three-Dimensional Computer Vision, The MIT Press, Cambridge, Massachusetts, 1993.

[7] G. Hager, W. Chang, A. Morse, Robot hand-eye coordination based on stereo vision, IEEE Control Systems (1995), 30-39.

[8] K. Hashimoto, editor, Visual Servoing, World Scientific Publishing, Singapore, 1993.

[9] R. Horaud, F. Chaumette, editors, Image-based Robot Servoing, special issue of International Journal of Computer Vision 37 (2000), 5-118.

[10] S. Hutchinson, G. Hager, P. Corke, A tutorial on visual servo control, IEEE Transactions on Robotics and Automation 12 (1996), 651-670.

[11] M. Lanzetta, G. Dini, An integrated vision-force system for peg-in-hole assembly operations, Intelligent Computation in Manufacturing Engineering, 1998, pp. 615-621.

[12] J.-C. Latombe, Robot Motion Planning, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1991.

[13] J. Pauli, Learning-Based Robot Vision, Springer Verlag, Berlin, Lecture Notes in Computer Science 2048, 2001.

[14] W. Press, S. Teukolsky, W. Vetterling, Numerical Recipes in C, Cambridge University Press, Cambridge, Massachusetts, 1992.