Taylor & Francis
Taylor & Francis Group

Flexible multi-sensorial system for automatic disassembly using cooperative robots

P. GIL*, J. POMARES, S.vT. PUENTE C. DIAZ, F. CANDELAS and F. TORRES

Department of Physics, Systems Engineering and Signal Theory, University of Alicante, Spain

Flexible multisensorial systems are a very important issue in the current industry when disassembling and recycling tasks have to be performed. These tasks can be performed by a human operator or by a robot system. In the current paper a robotic system to perform the required tasks is presented. This system takes into consideration the distribution of the necessary tasks to perform the disassembly of a component using several robots in a parallel or in a cooperative way. The algorithm proposed to distribute the task among robots takes into consideration the characteristics of each task and the sequence that needs to be followed to perform the required disassembly of the product. Furthermore, this paper presents a disassembly system based on a sensorized cooperative robots interaction framework for the planning of movements and detections of objects in the disassembly tasks. To determine the sequence of the disassembly of some products, a new strategy to distribute a set of tasks among robots is presented. Subsequently, the visual detection system used for detecting targets and characteristics is described. To carry out this detection process, different well known strategies, such as matching templates, polygonal approach and edge detection, are applied. Finally, a visual-force control system has been implemented in order to track disassembly trajectories. An important aspect of this system is the processing of the sensorial information in order to guarantee coherence. This aspect allows the application of both sensors, visual and force sensors, co-ordinately to disassembly tasks. The proposed system is validated by experiments using several types of components such as the covers of batteries and electronic circuits from toys, and drives and screws from PCs.

Keywords: Automatic disassembly; Cooperative robots; Recognition; Visual control; Force control

1. Introduction

During the last years the environment has become an important issue in the industrial world. The governments have created regulations to control different aspects of industrial production. In regards to this, the European Commission has introduced different directives to control 'The Waste of Electrical and Electronical Equipment (WEEE) in the early 2000s in its directives [2002/95/EC-2002/95/EC/2003/108/EC]. The law takes the environment into consideration along with the full life-cycle of a product: the design step, the useful life with the consumer and the product when it reaches the end of its useful-life time. This type of policy tries to reduce the great quantity of waste generated by a product when its useful life ends. This policy has great benefits for the environment (Williams 2006).

Ecological and economics can be some of the motivations to perform the recycling of products (Güngor and Gupta 2001). Ecological motivations are based on the environment maintenance, when a disassembly of products is performed (Bouikni *et al*. 2006). The environment receives benefits, thanks to the recycling of the toxic components of the product which can be separated to avoid contamination. The economical reasons to perform a disassembly of a product are based on the separation of

---

*Corresponding author. Email: Pablo.Gil@ua.es

the components and the raw materials that form a product to manufacture a new one (Gerner *et al.* 2005, Casper and Erik 1998). Furthermore, if the components are disassembled in a correct way, they can be reused directly (Kuren 2006). This would reduce the manufacturing cost of these components (Feldman *et al.* 1998, Eckterh *et al.* 1998).

As argued in (Eckterh *et al.* 1998), the disassembly is an essential issue in performing the recycling of electronic components. It allows one to:

(a) Remove toxic components. Before recycling any product the toxic materials must be removed from the product to avoid the contamination of the rest of the components. It is important to take into consideration the European Community law about the use and restrictions of use of toxic materials in electrical and electronic equipment (RoHS).

(b) Recover valuable materials. The components of the products can include expensive materials in their manufacturing. Their recovery is interesting for its ecological value as well as their economical value.

(c) Disassemble operative components for their reuse. The disassembly of the products permits the re-use of the components, improving the economical benefits and the re-use of expensive components, for example central processing unit (CPU), memory chips and power supplies (Torres *et al.* 2004).

In the electronic industry the initial responses to these trends were mainly of a defensive nature, directed towards compliance with legislation and towards preventing a negative image developing in the press and with the public. Emphasis was therefore on issues such as eliminating banned substances, cleaner production, recycling of packaging, and power management for the standby mode (Stevels and Rose 2001).

The system presented in this paper uses a multisensorial system as well as taking cooperative robots into consideration to perform disassembly tasks. This is one of its advantages, in contrast with the existing systems referenced in the bibliography.

In the current paper the different parts of our disassembly system are described. First the architecture used by the system is described. Second, the cooperative strategies used to optimize the disassembly system are described. Subsequently, in section 3, the strategies for the visual detection of pieces to be disassembled are presented. In section 4, the methodology for combining visual information obtained in section 3 and other information such as data obtained from a force sensor to help in the disassembly process, are also commented on. Finally, experiments of disassembly of screws, cover of batteries, and electronic circuits from products using the strategies and methodologies commented on this paper are shown.

2. Adopted strategies for cooperative movements

To perform the disassembly task a cooperative strategy has been adopted. This strategy uses several robot arms to do the desired task. This aspect permits a greater performance and gives more flexibility in a disassembly system.

The task planner schedules the tasks to perform a specific disassembly. These tasks are obtained with a precedence relationship among them. They can be represented like a tree. From each of these tasks, a group of rules are obtained. These rules are the specific parts that compose a specific task. Each rule represents a pair of components that has to be separate in a specific way to accomplish the task. Each rule can be divided in several actions and each action can be divided in sub-actions as well. Each action represents a specific movement to separate components or some part of them. Each sub-action represents an action that can be divided in small movements. The trajectory to be followed by the robot to carry out these movements is determined using our previous trajectory planning system or movement generator (Pomares *et al.* 2004).

Therefore, it is possible to emphasize two basic steps. On the one hand, the planning of the tasks to accomplish the disassembly process, as well as the distribution of these tasks among the robots, are determined. This part is described below, in the following section. On the other hand, it is necessary to generate the movements as well as the planning of the trajectories that must be executed by the robot to carry out an action of the disassembly process (Pomares *et al.* 2004).

2.1. Architecture of our system

In figure 1, the architecture of the system is shown. It is composed of two robots arms, one PA-10 from Mitsubishi and one Scorbot ER-IX from Intellitek. The first robot has seven degrees of freedom and the second one has five.

The robots have a force-sensor in their end-effectors, a tool-changer and a mini-head camera to perform detection of objects and visual servoing to track trajectories. The PA-10 robot has several tools to perform the different tasks: a two-finger parallel pneumatic griper, a vacuum, an electric screwdriver and a cutting tool. The Scorbot robot has a two parallel fingers electric gripper.

2.2. Task planning

The disassembly strategy is based on the hierarchical model proposed in (Torres *et al.* 2003), which uses a graph representation of the product in which the relations among components are expressed by means of arrows. With this model, the relation existing between two components is represented with an arrow between the two nodes that

symbolize the components, and the set obtained can be considered as a whole subassembly, which can be included as a new individual component in the model.
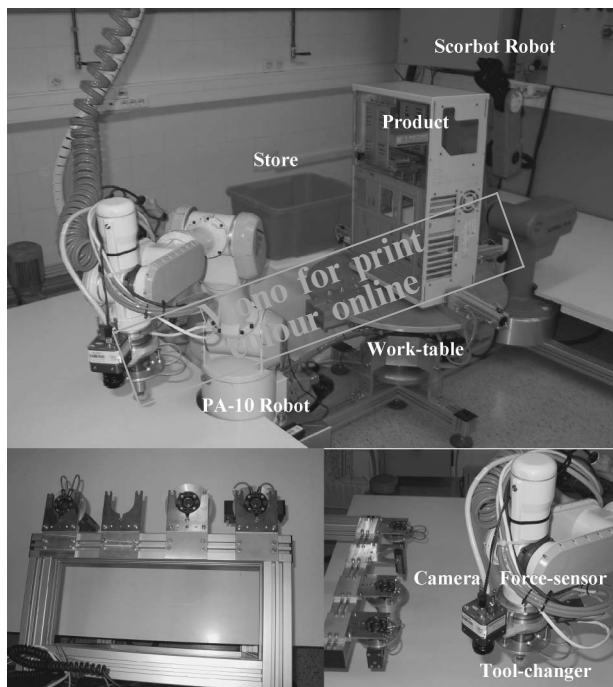

Figure 1.   Architecture of the system.

Using that model the task planner can determine the sequence of components that must be removed to achieve a specific sequence of tasks. If the target consists of the disassembly of a specific component, the task planner can provide the best sequence to reach the specific component (Torres *et al.* 2003). If the target is the complete disassembly of the product, the task planner provides the best sequence completely to disassemble the product (see figure 2).

Based on the sequence of tasks provided by the task planner, a group of rules are determined. Each task can be divided in one or more rules to disassembly a specific component. With these rules a tree structure is created, which determines the order that must be followed to achieve the target in the disassembly planning.

Afterwards, when the task planner has determined the tree of actions, this information is used to distribute the actions among the robots that have to do the task (see figure 3).

In figure 2 a tree with the tasks to be performed to completely disassemble a toy are shown. First of all, the screws which hold the battery cover and the plastic cover have to be removed. This task can be divided into the removing of the screws that hold the battery cover and the removing of the screws which hold the plastic cover. After taking out the screws of the battery cover, the cover can be removed and then the batteries will be accessible to be removed as well. On the other hand, after removing
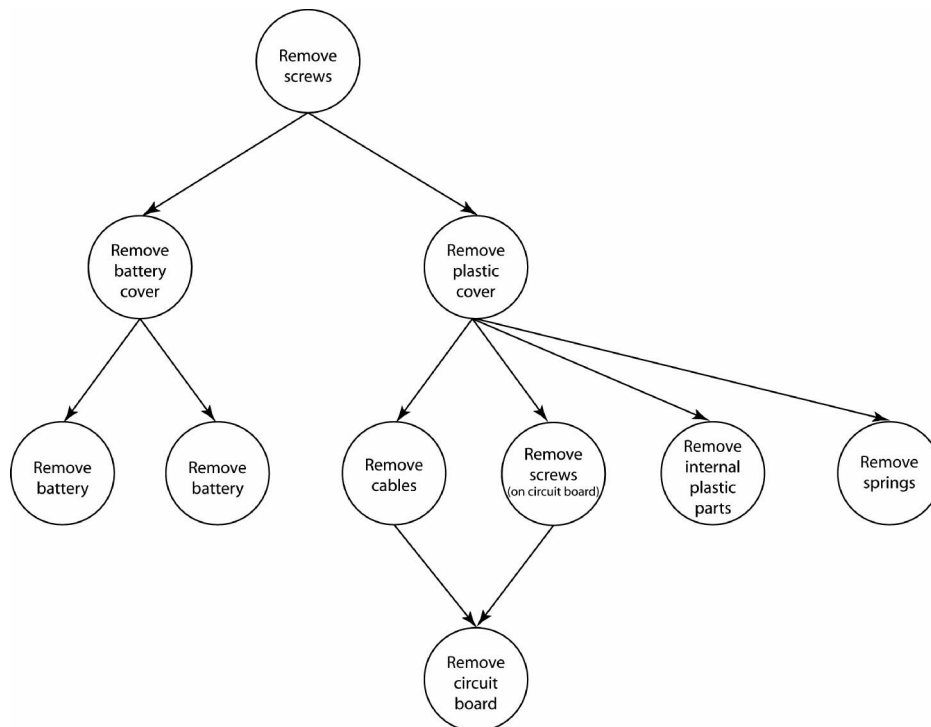

Figure 2.   Tree with the tasks to disassembly a toy.

the screws which hold the plastic cover, the plastic cover can be removed. After removing the plastic cover, the toy is open showing the circuit board, the different internal plastic components, and the springs to move the parts of the toy. Following the sequence, to remove the circuit board two actions need to be performed in this specific example. These two tasks can be performed in parallel, at least theoretically, but in practice they must be performed sequentially because the size of the workspace is limited and it is impossible to employ the two different needed robot tools at the same time. These tasks are the removal of the power cables of the circuit board and the removal of the screws that hold the circuit board.

2.3. *Distribution of the task among the robots*

The initial point is a tree with the tasks to be performed [see figure 3(a)]. Each of these tasks can be divided in actions and sub-actions that represent the smallest part of a task that include information about the tool to be used. Next, a tree of actions and sub-actions to disassemble a personal computer (PC) product is shown and commented on [see figure 3(b)].

Using the tree of tasks [see figure 3(a)] a decision-tree [see figure 3(b)] is created taking into consideration the information on each task as well as the information about the existing robots in the working cell. Furthermore, about each robot it is important to know its location in the working cell, its working area, its payload, etc. These characteristics will determine if the robot can perform an action because the product to be manipulated is inside of its working-area or because it is the optimal robot to be used.

On the one hand, the algorithm considers a working table, that can be rotated, exists in the working cell to distribute the task among the robots. This gives the system one degree more of freedom, allowing the location of the products in the working area of a specific robot and to perform the desired manipulation using the best orientation. On the other hand, the algorithm takes the different tools required to perform the tasks into consideration. To do this, the inclusion of a tool-changer for each robot is implemented in the algorithm.

For distributing the actions among the robots, the previous items are considered in the following order. Each robot has a different cost that is used to choose the
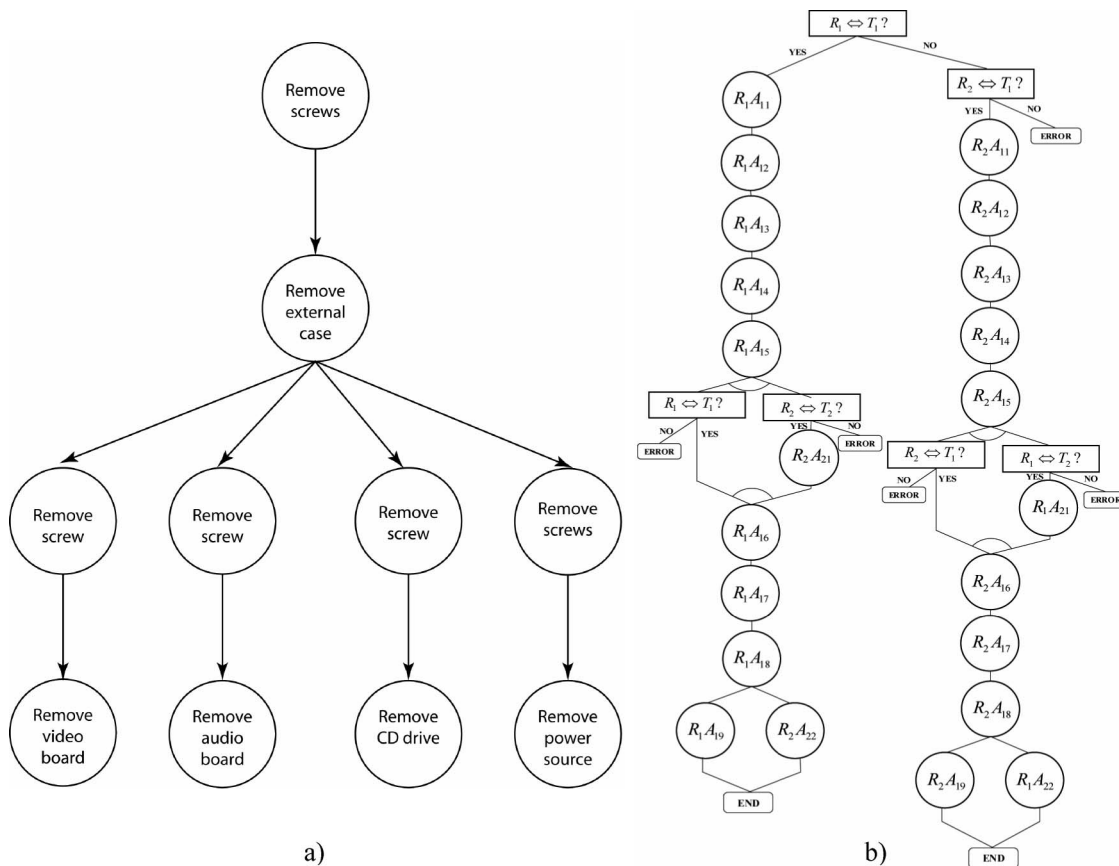

Figure 3.   Disassembly of a PC. (a) Tree with the tasks to disassembly a PC. (b) Tree with the decision-tree of actions to perform by each robot to remove a CD drive from a PC.

adequate robot for performing a certain action depending on its physical specifications. The rotation of the working table also has a cost. And when it is necessary to change the tool using the tool-changer a greater cost is included for the task. Finally, the cost of doing a specific action is included. The robot that provides a lower cost for a specific action is designed to perform that action.

First, figure 3(a) shows the tasks to perform the disassembly of a PC. First of all, it is necessary to remove the screws that hold the external case, after this the case can be removed and then there are several tasks that can be performed in parallel. The quantity of parallel tasks will vary according to the specific configuration of the PC. In this example, each component is held with some screws, and the components are: a video board, an audio board, a CD drive and a power source. Once this tree of actions is obtained, the tasks are distributed among the robots, according to the previous algorithm.

In figure 3(b) the decision-tree for distributing the actions among robots is shown for the task of removing a CD driver. In this tree the tasks are divided into actions to be accomplished, and with these small actions the parallelization of the actions is computed and each action is assigned to a specific robot to be performed. In that tree, the labels R_1 and R_2 represents the robot 1 or 2 to be chosen to perform the desired action. T_i represents the tool needed for the action i. In addition, the actions are represented as A_{ij} where i represent the number of the action and j the sub-action. In this example the actions A_{11}, A_{12}, A_{13}, A_{14}, A_{15}, A_{16}, A_{17}, A_{18} and A_{19} represent respectively the following actions: remove the first screw, leaving the first screw in the store, remove the second screw, leaving the second screw in the store, rotating 180° the work-table, remove the third screw, leaving the third screw in the store, remove the fourth screw, leaving the fourth screw in the store. And the actions A_{21} and A_{22} represent holding the CD driver and removing the CD driver respectively. Therefore, first the robot 1 removes the screws that hold the CD driver in one side of the CD driver and leaves them in the store (A_{11}, A_{12}, A_{13}, A_{14}). After this, the PC is rotated (A_{15}), using the work-table, and then the other two screws can be removed, but previously it is necessary to use the other robot to hold the CD driver (A_{21}) to avoid it crashing when the screws are removed (A_{16}, A_{17}, A_{18}, A_{19}), and in parallel to leaving the last screw in the store (A_{19}) the action of removing the CD driver is performed (A_{22}).

3. Adopted strategies for visual detection of targets

3.1. Target and characteristics detections

The strategies used for visual detection of targets can be divided into two categories. The techniques of the first category use grey values and recognition based on patterns already known. The techniques of the other category use contour characteristics to determine a bi-dimensional geometry model from local structures in terms of edges.

The objective of our strategies is to recognize the type of component to disassemble and fix its image position (Torres *et al.* 2004). On the one hand, from the image position and from the knowledge of the kinematic and transformations of each element that composes the disassembly cell, the position of each component is computed. Thus, the initial position of the image is used to guide the robot and its tool until the desired position. On the other hand, the knowledge of the type of component defines the tasks set for planning its disassembly operation.

Owing to the industrial environment of the selected scenario, there are variety of objects and components. Therefore, we have taken special interest in the disassembly of components for example screws, covers, batteries, chips and electronic circuits, and wires. The screws and wires are very important to separate the components in order to disassemble batteries, covers, chips and electronic circuits. These objects are important from a point of view of recycling. In recent years, a number of researchers have focused the vision systems and the recognition process on the disassembly tasks. Examples of these systems are, a demanufacture application of mobile phones based on structured lighting to estimate the product surface (Bailey and Kuren 2006), and a vision-guided robotic system for an autonomous disassembly process of wheels (Büker *et al.* 2001). In this last work, a combination of contour and PCA-based recognition is used. ②

3.1.1. Characteristics detection.
Two of the main problems in characteristics detection are: the instability problem and the time of computation to extraction of characteristics. The first problem makes it difficult when trying to recognize real objects in industrial surroundings, since slight illumination changes or viewpoint can change the luminance values in image and modify the detected characteristics. Moreover, an object can hide part of a portion area of another object making more difficult the characteristics extraction process (Gil *et al.* 2006). It is the know-problem of occlusions in the assembled components of an object. In this way, sometimes, the searched interest characteristics are not found and other times the found characteristics are not interesting. The second problem can produce important delays in the guidance of the robots and increase the run time of the planned tasks. Therefore, the extraction of characteristics has to be as fast as possible.

These factors determine the election of the information about the contour of objects, like a type of characteristics that it is necessary to extract. The contour of objects is computed from Canny's detector (Canny 1986) or an adaptive thresholding, depending of the type of component

to detect according to the graph of planning tasks. Moreover, the smoothing process is important to reduce the effects of noise when the partial derivatives are calculated to determine interest points in the image. Hence, noise is removed by convolving the image with a rotationally invariant Gaussian mask. After obtaining the contours of the object to disassemble by means of one of these two techniques, the next step is to determine the edge points like the interest points which define the contour.

Finally, the Douglas–Peucker's algorithm (DP) is used to obtain a polygonal approach of the edge points and the progressive probabilistic Hough transform (PPHT) to determine the lineal approach of those same edge points. In the same way as previously, the election of one approach method or another, lineal or polygonal, depends on the type of component that needs to be detected.

Independently of the chosen approach, DP or PPHT, a post-processing is necessary. For example, sometimes we have to cluster groups of the different lines detected using orientation and distance information (i.e. in PPHT). Another example is the hierarchic classification of the contours and later polygonal approaches (i.e. in DP).

DP's algorithm reduces a set of many interest points detected to a simplified form. This is useful in situations where the extraction of interest points is not very stable. The linear and polygonal forms are more stable than the points.

To illustrate, the detection process with the previous technical comments, figure 4 shows the detection of screws for the disassembly of an electronic circuit. The captured image is processed by adaptive thresholding to extract contour information. Afterwards, the contour image is used to extract contour points. Finally, the polygonal fitting of the contour points with DP algorithm permits to obtain the candidate region of screws.

Figure 5 shows the detection of wires for the disassembly of the circuit. First, the Canny's detector extracts contour information. Second the pattern of the circuit is recognized and the position in the image determined. And finally, the PPHT algorithm is used to detect interest lines within the delimited region by the circuit. These candidate-detected lines, which represent parts of the wires, are post-processing to obtain an approximated location of the wires. The post-processing proposed reduces the lines by clustering the lines of similar orientation and proximity factor, and simultaneously it is possible to compute the orientation and position for the cutting tool. The orientation for the cutting tool is given by the orientation of the final lines detected and the position is calculated in the middle point of the length of these lines.

3.1.2. Region detection. The techniques used for region detection are based on pattern recognition by means of matching templates. The key factor is the good behaviour for real-time applications so that these techniques are adapted to this type of disassembly applications. The techniques for region detection are mainly employed for the recognition of components. When we talk about region detection, we make reference to the place or situation in the image plane where a component is located. If a component is in the image, it is to say that the recognition process is positive, so the position and orientation in the image is computed.

The components for disassembly have been modelled previously and they are known. In order to decrease the complexity in the process of recognition, a few and simple features are stored as knowledge base. It is known that the greater number of features, the greater running delay. The stored features have been extracted with a previous off-line pre-process. These features are extracted from a process of contour extraction and average values of luminance in neighbouring surroundings. In order to avoid instability, the off-line pre-process is realized with similar illumination conditions to those of on-line process. The

a)   b)   c)
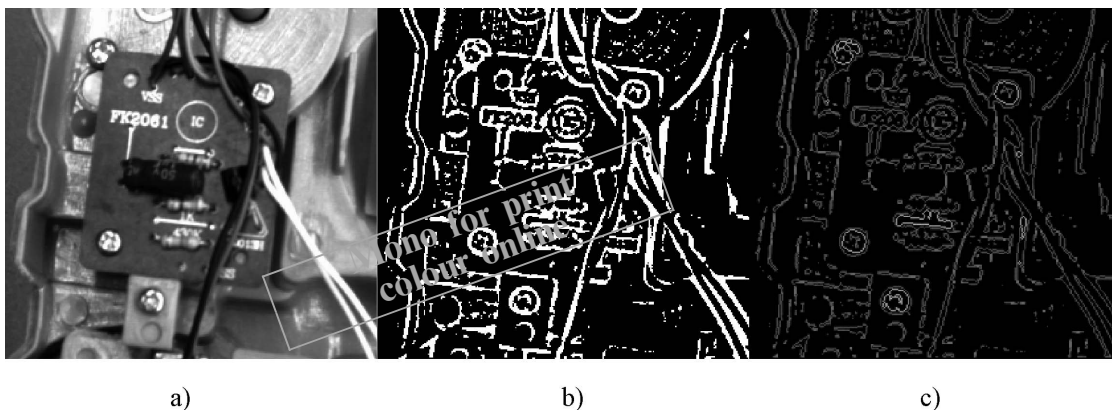
Figure 4.   Detection of screws. (a) Original image of electronic circuit. (b) Detection of contours. (c) Polygonal fitting and detection of regions of screws.

off-line pre-process has been made for every component to recognize.

For example, in one of the experiments that we have made, a set of batteries within a toy was disassembled. To do this, the compartment that kept the batteries has to be recognized and detected in the image. The compartment is detected from the recognition of the cover that closes it. This cover is modelled from the borders that form its contours because they define a geometrical form. It is computed during the off-line pre-process. Therefore, the extracted contours are the principal features to detection.

Later, during the on-line process, the identical methodology of the off-line is made. Summarizing, the vision system takes images of scenes with products (i.e. the toy),

and for each image, an extraction process of contours by Canny's detector is made. Then, some geometrical features are obtained from the contours, such as lengths, orientation and the distributions of them. The next step is matching the process between the extracted features by means of off-line and on-line processes. With this process, the similarity between both set of features are compared. The similarity is measured as a probability factor. If this one is superior to the threshold of tolerance, the component is considered found. When this happens, the region of the image where it is found is computed as a position, an orientation and area. This information will be used by the planner so that the robot can select the suitable tool (i.e. the cupping glass) and later to disassemble the component (i.e. the cover) in a suitable way. Figure 6 shows the detection of the cover that closes the batteries of a toy and the detection of the circuit.
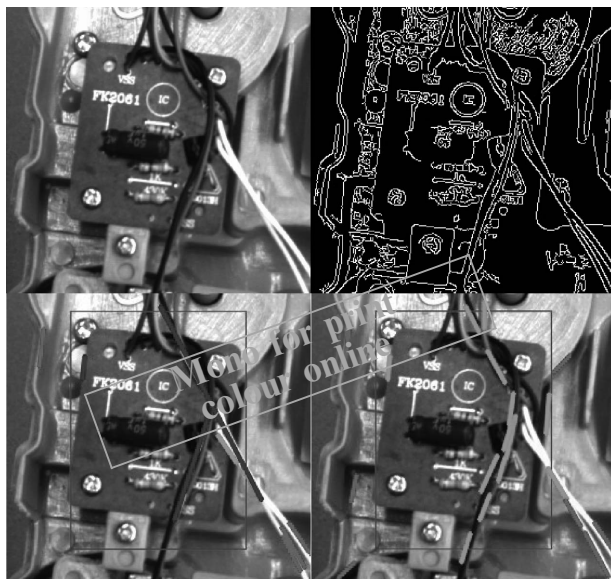

Figure 5. Detection of wires. (a) Original image of electronic circuit. (b) Detection of contours. (c) Recognition and location of the circuit from image. (d) Detection of lines which determine wires.

3.2. Geometry and mathematical principles of the strategies

In this section, geometry and mathematical principles used to target the detection and characteristics are described briefly.

3.2.1. Preprocessing. Thresholding is a well-known technique for image segmentation. Thresholding is the operation of converting a multilevel image into a binary image $f_T(x, y)$. This method is used, jointly with Canny's detector, as a pre-processing step to extract the contours of pieces and its components from the image. These methods are employed, in the first step because it permits a significant reduction of characteristics to analyse. The principal disadvantage of standard thresholding techniques is the necessity to fix the threshold parameter T. In real disassembly applications, rarely, a threshold fix can be used for all the regions of the image. This fact is the consequence of the illumination variations and irregularities of the capture devices that produce local variations in the grey level of the pixels of each region. For this reason, here it has been resorted to a variant called adaptive
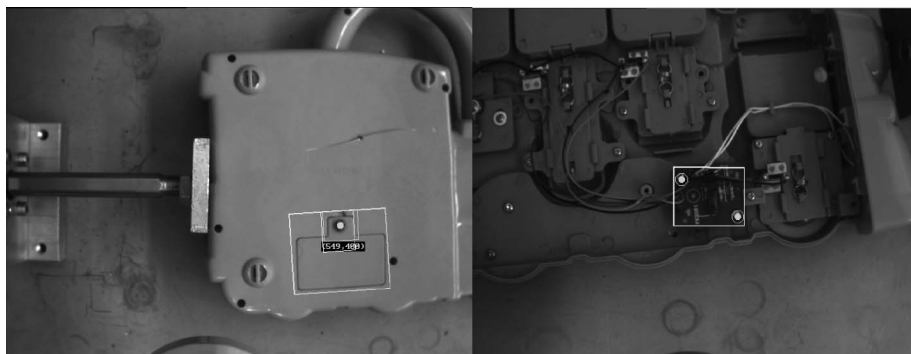

Figure 6. Detection of a cover of batteries and circuit with the location of its screws.

thresholding. In this variant of thresholding, the threshold parameter is not previously fixed because it will depend on the position of each pixel in the image. This way, each region of the image, or even each pixel is processed in respect to a threshold calculated in local surroundings of the neighbourhood. A simple and fast way to compute an adaptive threshold T statistic information is used, such as the mean of the local intensity distribution. This method is known by adaptive thresholding by mean-C where C is the size of neighbourhood mask chosen. In this case, we have employed a mask 9×9.

$$f_T(x, y) = \begin{cases} 255, & \text{if } f(x, y) \geq T \\ 0, & \text{if } f(x, y) < T \end{cases} \text{ where } T = \text{mean} - C \tag{1}$$

3.2.2. Douglas Peucker's polygonal approach.

The idea principal of DO (Douglas and Peucker 1973, Hershberger and Snoeyink 1992) is to simplify the polyline which fits a contour. A polyline is defined as an approach by means of segments or polynomials of degree one. The simplification is based on representing a set of points by an edge segment where the closeness of each point to the edge segment has to be to the inferior to a tolerance factor taken as reference. The closeness is measured as a normal vector of distance to the candidate edge segment. The algorithm starts approximating all the points of a same contour by means of a polyline that join the first and last points. The distance from each intermediate point to the polyline is measured, and the points furthest from the polyline with a tolerance greater than zero is added the new polylines simplification form. This point will divide the previous one in two new polylines. Thus, this algorithm continues iteratively for each contour until all points of the polylines are within tolerance of the simplification. If the distance of an intermediate point is less than the tolerance, this one is eliminated.

3.2.3. Progressive probabilistic Hough transform.

Another principle used is the PPHT. This is used for lineal approach. This approach has interest when we want to detect components with large perimeter and little area, for example to determine the position of wires in images. The PPHT is a class of Hough transform (HT). The HT is a popular method for the extraction of geometric primitives and PHT is a version of HT that introduces the principle of randomness. Thus, the probabilistic Hough transform

(PHT) is a probabilistic class of HT where only a pre-selection of input contour points are used, and PPHT is a variant of HT based on the PHT. PPHT algorithm is very suited for real-time applications with limitation of the processing time. Next, we describe briefly, HT and its variant PPHT work.

The aim of HT (Illingworth and Kittler 1988) is to detect lines $\{l_i\}_1^n$ in an image f(x, y) of contours where each contour is constituted by a set of points $\{p_i^{C_j}\}_1^m$, called contour points and defined as $(x_i, y_i)_{C_j}$. The lines detected are defined by two parameters ρ and θ, such that they define the line parametric equation:

$$l_i(\rho, \theta) = x_i \cos(\theta) + y_i \sin(\theta) - \rho \tag{2}$$

The HT considers ρ as the distance from the origin to the line, and θ is defined as the angle from the x-axis to the perpendicular to the line vector which points from the origin to the line. In order to obtain the lines, $\{l_i\}_1^n$, an accumulator matrix, $A(\rho, \theta)$, is defined. Initially, this accumulator contains only zeros. The process is based on one increment by all of the accumulator elements that correspond to a line which contains the point. Each increment is called vote and defines a possible line determined by ρ and θ. The algorithm repeats this process for each contour point. In addition, a threshold is defined to distinguish lines and noise characteristics. This threshold determines a number of minimum votes so that a possible line can be considered line detected. The variant PPHT employed (Matas *et al.* 1998, Galambos *et al.* 1999) minimizes the amount of computation needed to detect lines with different numbers of supporting points, thus it has the advantage that it does not need *a priori* knowledge about the lines lengths in the image. Furthermore, PPHT and HT performance only differ in the case of assignment of points to lines where points are in the neighbourhood of more than one line.

3.2.4. Matching template.

The main function of pattern recognition is to search for occurrences of a model, m(x, y), in an image, f(x, y). The mathematical base used is the normalized greyscale correlation. This technique is widely used in the industry for pattern matching applications. The correlation operation is a form of convolution, where the convolution kernel is defined by a pattern matching model. In order to reduce the influence of brightness of each pixel, the solution is to use a normalized version.

$$r = \frac{N \sum_{ij} f(x,y)m(x,y)^2 - (\sum_{ij} f(x,y)) \sum_{ij} m(x,y)}{\sqrt{\left[N \sum_{ij} f(x,y)^2 - \left(\sum_{ij} f(x,y)\right)^2\right] \left[N \sum_{ij} m(x,y)^2 - \left(\sum_{ij} m(x,y)\right)^2\right]}} \tag{3}$$

N : number pixels of the model

In addition, if the technique of resolution pyramid is incorporated, it is possible to reduce the time of computation for searching (Manickam *et al.* 2000). The solution is to reduce the resolution of image or/and model and to search hierarchically. First, the model with little resolution is searched, and several match candidates are founded. After, the match candidates are refined by means of an iterative process increasing the resolution in each step or iteration.

Later, areas of image of equal size are compared by sum of squared differences (SSD) between pairs of corresponding pixels. And the sum is divided by the number of pixels compared to give a final match probability range of zero to one.

$$P = \frac{SSD_{ij}}{N} = \frac{\sum_{ij}(f(x,y) - m(x,y))^2}{N} \tag{4}$$

N : number pixels of the model compared

4. Combining force and visual information for disassembly

4.1. *Tracking trajectories by using an eye-in-hand camera system*

Using the disassembly system, described in section 2, the path to be followed for the removal of a given component is determined. Once the path has been defined, it must be followed by the robot employing an eye-in-hand camera system. When the robot is following the path, unexpected problems may arise, such as an unknown contact element (i.e. welding), the appearance of wires obstructing the trajectory, the gripper not being perfectly perpendicular to the grasping point, and so on. Therefore, some reactive system has to be used to perform the correct disassembly task in order to avoid such unexpected problems. To do the tracking of trajectories, a set of M points are used. These points are computed from characteristics in the process of object detection described in section 3. Therefore, these points depend on the type of component disassembled. For example, points of the contour are used for disassembly of a cover, and points of line which approximate the geometrical form are employed for disassembly of a wire.

Employing the demanufacturing system previously described, the desired trajectory in the 3D space to be tracked, is generated. Sampling this trajectory the set $\tau = \{{}^k p_o^X / k \in 1, \ldots, N\}$, is obtained, in which ${}^k p_o^X$ is the location of the object O with respect to the coordinate frame X at the instant k (the trajectory being specified with respect to the object O). ${}^k p_o^X$ is projected in the image at the instant k at a point of homogeneous coordinates ${}^k m_o^X = \begin{bmatrix} {}^k x_o^X & {}^k y_o^X & 1 \end{bmatrix}^T$. The corresponding coordinates in pixels will be denoted as ${}^k f_o^X = \begin{bmatrix} {}^k f_{xo}^X & {}^k f_{yo}^X & 1 \end{bmatrix}^T = A \cdot {}^k m_o^X$

where A is a non-singular matrix which contains the intrinsic parameters of the camera being used:

$$A = \begin{bmatrix} f \cdot p_u & -f \cdot p_u \cdot \cot(\theta) & u_0 \\ 0 & f \cdot p_v / sen(\theta) & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5}$$

where u_0 and v_0 are the pixel coordinates of the principal point, f is the focal length, p_u and p_v are the magnifications in the u and v directions respectively, and θ is the angle between these axes.

Considering a total of M points which are obtained from the object O, the discrete trajectory in the image space is also composed of M trajectories, $C_i / i \in 1, \ldots, M$ (one for each feature extracted from the image that corresponds to the M points). As such, the values of each of the M trajectories in the image, at instants k, are $C_i = \{{}^k f_i / k \in 1, \ldots, N\}$. If the set of M features is considered as a desired configuration in the image, the set S is obtained, such that $S = \{{}^k s / k \in 1, \ldots, N\}$, where ${}^k s$ is the set of M points or features observed by the camera at instant k, ${}^k s = \{{}^k f_i / i \in 1, \ldots, M\}$.

Each sample, ${}^k s$, is generated from each position, ${}^k p_o^X$. These positions are obtained considering that the time between two consecutive samples is constant, so that $\Delta^{k+1} t = {}^{k+1} t - {}^k t = T$ where T is the video rate. The desired trajectory to be tracked in the image is obtained using a natural cubic B-spline:

$$s_d(t) = {}^k A t^3 + {}^k B t^2 + {}^k C t + {}^k D \tag{6}$$

where ${}^k A$, ${}^k B$, ${}^k C$, ${}^k D$ are obtained from the samples in the image space at the given instants.

To perform the tracking of the desired trajectory in the image space an image-based control scheme to regulate to 0 the following vision-based task function is used (Mezouar and Chaumette 2002):

$$e = \hat{J}_f^+ \cdot (s - s_d(t)) \tag{7}$$

where **s** are the extracted features from the image and $\hat{J}_f^+$ is an estimation of the pseudoinverse of the interaction matrix (Chaumette 1998). To carry out the tracking of the trajectory, the following velocity must be applied to the robot (with respect to the coordinate frame located in the eye-in-hand camera):

$$v^C = -k \cdot e + \hat{J}_f^+ \cdot \frac{\partial s_d(t)}{\partial t} \tag{8}$$

where k > 0 is the gain of the proportional controller.

4.2. *Fusing force and visual information in the disassembly system*

To assure the correct development of a given task in which an interaction with the setting is required, the system must

carry out a variation of the trajectory in the image, depending on the spatial restrictions imposed by the interaction forces. Therefore, given a collision with the setting and recognized the normal vector of the contact surface, the transformation T that the camera must undergo to fulfil the spatial restrictions, is determined. This transformation represents the nearest direction to the one obtained from the image-based control system, and which is contained in the plane of the surface. Thus, we guarantee coherence between visual and sensorial information from the force sensor. To do so, considering f to be the position of a given feature extracted by the camera at a given instant, and $[R_i\ t_i]$ a sampling of the transformation T that the camera undergoes during the tracking of the recognized surface, the feature f'_i extracted at each one of these positions will be:

$$f'_i = \mathbf{A} \cdot \mathbf{R}_i \cdot \mathbf{A}^{-1} \cdot f + \mathbf{A} \cdot \mathbf{t}_i / z \qquad (9)$$

where $\mathbf{A}$ is the intrinsic parameter matrix [equation (5)] and z is the distance between the camera and the object whose features are extracted from the image.

From the sampling of the desired trajectory in the image, f'_i, a spline interpolator is applied to obtain the desired trajectory in the image.

Once the coherence between both sensorial systems is guaranteed the system can use the visual and the force information coordinately to track a disassembly trajectory. Now, the different phases that compose the system for fusing visual and force information are described. These steps are shown in figure 6 as an insertion task. In this task, the robot must maintain the contact with a constant force being applied between the tool at the end-effector and the surface. The previous mentioned visual servoing approach is used so that the desired image trajectory for each feature, f_{id}, is the one observed by the camera during the trajectory of the robot (maintaining the contact with the surface). However, this trajectory is subject to errors and, therefore, must be modified according to the information obtained from the force sensor.

Once the contact between the tool and the object is detected [see figure 7(a)], the system recognizes the contact surface, correcting possible errors in the orientation of the robot with respect to the object [see figure 7(b)]. Once this step has been finished, the system determines the direction nearest to the one obtained from the visual servoing system which fulfils the spatial restrictions imposed by the manipulated object. Once this direction is obtained, the image trajectory, tracked by the visual servoing system, is modified, as was previously described. Thus, the system is able to track the trajectory, maintaining the contact with the surface and obtaining compatible control actions from both sensory systems [see figure 7(c)]. Furthermore, we have developed a system for detecting changes in the model considered by a Kalman filter applied to the interaction forces obtained (Pomares and Torres 2005). This system is used not only to detect changes in the surface but also to fuse the visual and force information.

The final control action, v^C, will be a weighted sum obtained from the visual servoing system, v_M^C, and from the force sensor, $v_F^C = \lambda_F \cdot (F - F_d)/k$ (where F and F_d are the obtained and the desired interaction forces, k is the tool stiffness, and λ_F is the propotional control gain for the force controller), so that $v^C = p_v \cdot v_M^C + p_F \cdot v_F^C$.

5. Results

5.1. *Experiment 1. Screw disassembly*

Now the proposed system is tested for the removal of a screw from a rail. To carry out this task, the experimental setup shown in figure 8 is used. First of all, the hierarchical model of the product, composed by the rail and the screw is modelled. The hierarchical model is used by the task planner to determine the sequence of the components that need to be separated to acquire accessibility to the desired one, the screw. In this case the component is directly accessible, owing to the characteristics of the product that we are working on. Once the component that needs to be separated has been chosen, the trajectory to be performed has to be computed using the movement generator. In this case the trajectory obtained is to move the screw in a perpendicular direction to the rail and orientated to the top surface of the rail (see figure 9).

a)   b)   c)
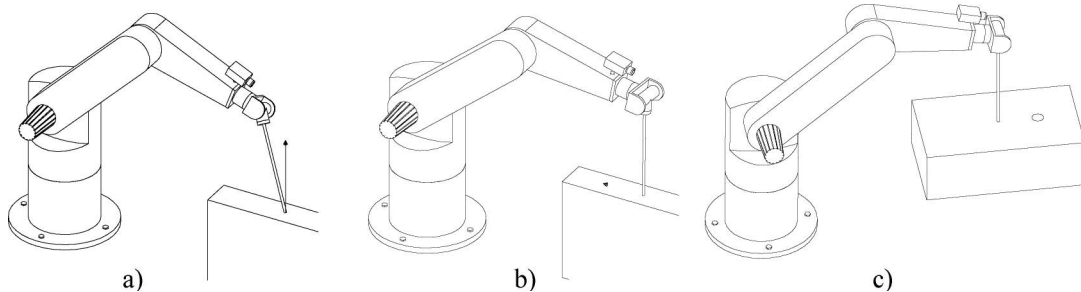
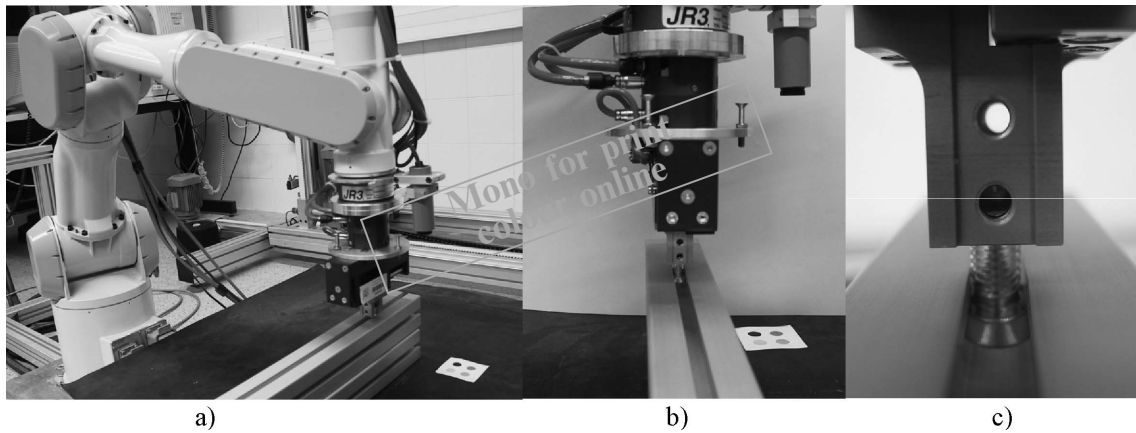Figure 7.   Steps of visual-force control strategy applied to an insertion task.

Figure 8.   (a) Experimental set-up. (b) and (c) Detail of the disassembly process.

Considering this path to be followed by the screw, the robot detects a collision between the rail and the screw when the real task has been performed. To correct this situation a simultaneous control of the position and of the interaction of the robot has to be performed.

Figure 9 shows the different phases required for the disassembly. The removal direction obtained by the movement generator is F_o [see figure 9(a)], that can be used to move the screw from the initial position to the top surface of the rail. When the screw collides with this surface, the forces F_r and $F_{r'}$ appear in the opposite direction of F_o [see figure 9(c)], these forces must be avoided to remove the screw. To achieve the correct removal of the component, the new direction F_e is generated for removing the component from the rail. This new direction can be seen in a perpendicular view of the rail in figure 9(d).

Obviously, when the disassembly is being done in the direction indicated in figure 9(a), the head of the screw will eventually collide with the rail, so that, the disassembly will no longer be possible. This happens if the screw is not perfectly modelled, i.e. it is modelled as a cylinder but the head of the screw is not considered.

The geometric model employed by the task planner commented in section 2 is not the perfect model of the component. This simple model also allows an easy way to perform the generation of the trajectory computed for the disassembly of the component by the movement generator.

In such a case, the situation shown in section 4.2 arises. That is, there is a contradiction between the direction established by the vision system and the interaction information obtained from the force sensor. In accordance with the algorithm previously presented, and once the contact surfaces have been recognized, the next step will be to modify the trajectory in the image, using the information from the force-sensor, so that the new movement direction will be coherent with the interaction information obtained.
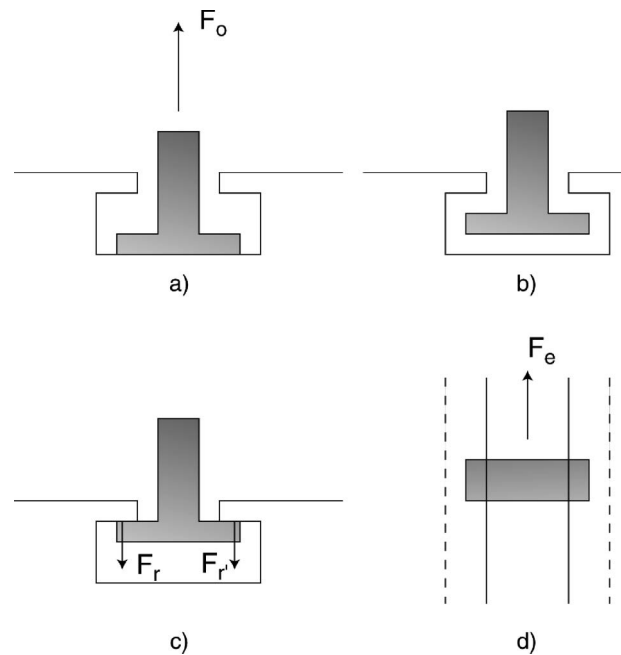

Figure 9.   Removal of a screw from a rail.

In this task, the screw in the rail shown in figures 8(b) and 8(c) is to be removed. To carry out the removal of the screw, its head must maintain a constant force against one side of the rail during the disassembly (1 N. in x direction), and also against the upper part (5 N. in z direction). Once the robot has grasped the screw and the contact surface is recognized, the desired trajectory in the image is generated to carry out the disassembly process. This trajectory is used by the image-based control system to develop the disassembly trajectory.

In figure 10, the evolution of the forces obtained in the joint use of the force and visual control, is shown. In this figure, we can observe, that once the forces are established at their desired levels, the displacement of the screw along

the rail is carried out. In the last phase of the graph, the forces are quickly annulled and, therefore, the disassembly process is finished.

5.2. *Experiment 2. CD drive disassembly from PC*

In the second experiment the disassembly of the CD-drive from PC is made (see figure 11). In this experiment, we have put in practice the cooperation between two robots. Perhaps, sometimes, in the disassembly of a product (i.e. PC), the cooperation between two robots is necessary to realize tasks like removing the CD-drive. The tree of tasks in this experiment was shown in figure 3, and in section 2.3 it was described how the robot 1 (Scorbot) must hold the
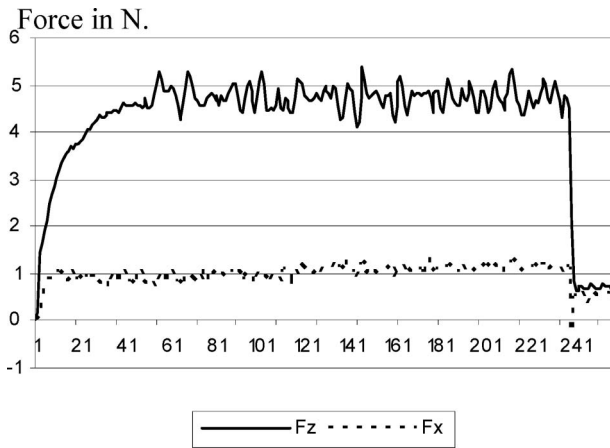

Figure 10.   Evolution of the generated forces during the disassembly process.

CD driver while the robot 2 (PA-10) removes the screws which hold the CD driver to the internal case of the PC. If in this task the robots did not work jointly, the CD drive would fall outside of the case. In the fall it could damage the rest of the components of the PC making it difficult for the next disassembly task.

5.3. *Experiment 3. Cover of batteries disassembly from toy*

In this experiment the disassembly of the cover of batteries is developed (see figures 12 and 13). Here, the methodology used to detect the cover and its screws has been commented on subsection 3.1.2. As in the previous experiment the disassembly direction computed to movement generator is determined by a previous model of the product to be disassembled and therefore this direction can have errors. From the model, the disassembly direction computed is only upwards, however in order to make the adapted disassembly it is necessary to develop the movements illustrated in figure 14. It is therefore necessary to modify the previous generated trajectory in order to be coherent with the interaction forces obtained from the force-sensor at the end of the robot.

When the disassembly begins tracking the previous determined direction, torques appear due to two hooks which fix the cover. First, this trajectory is tracked only using visual information. However, it is necessary to modify the visual information depending on the torques which appear during the tracking. The torques are compensated and the rotation movement shown in figure 14(b) is obtained. In order to be able to control the torques

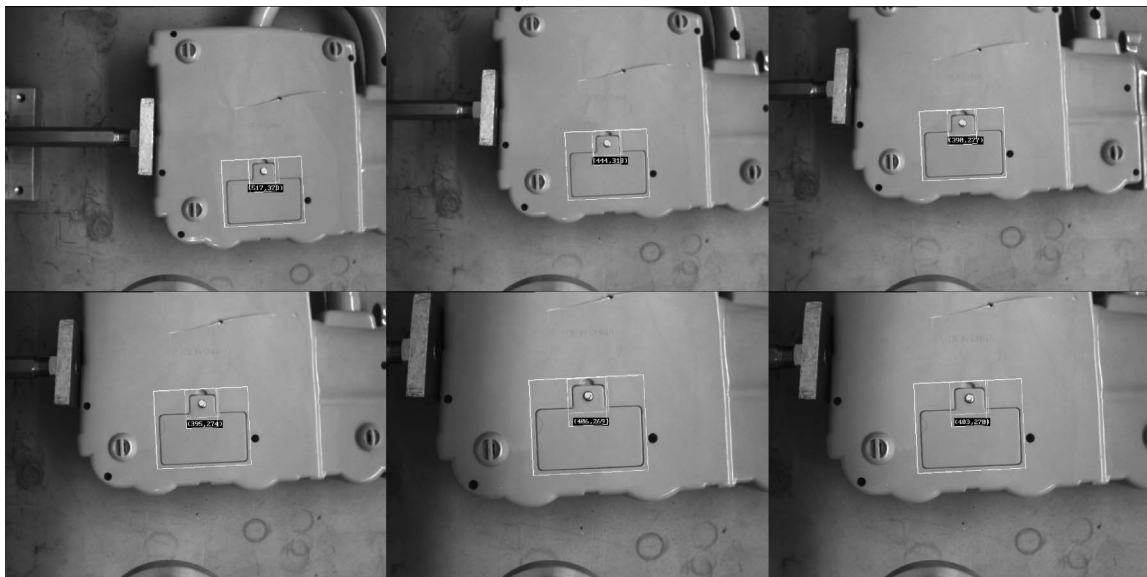Figure 11.   Sequence of movements in the disassembly of CD drive.

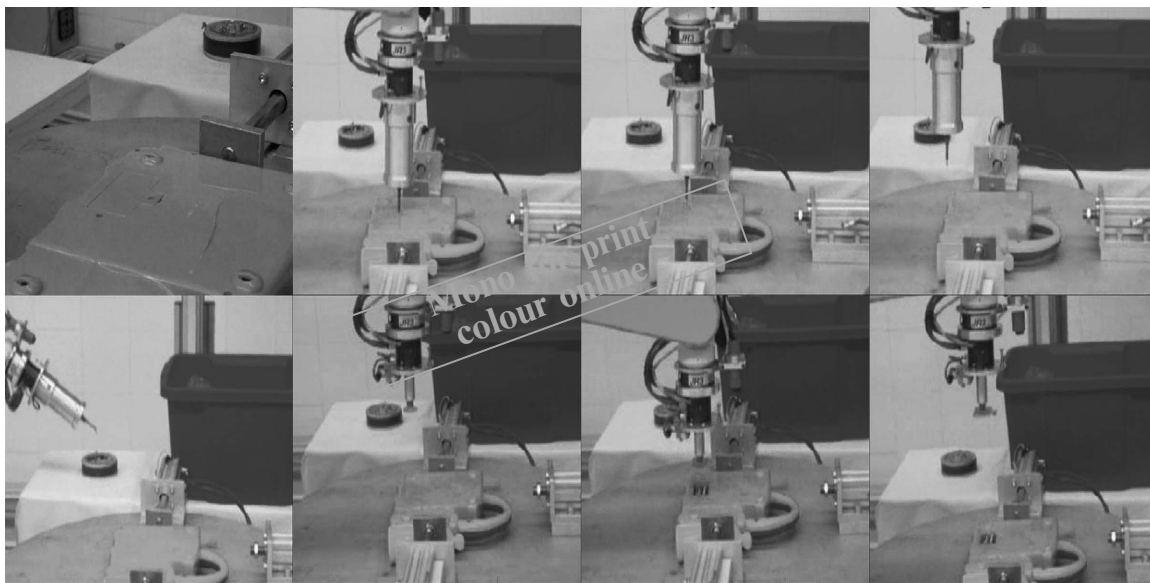Figure 12.   Sequence from eye-in-hand camera to disassemble the cover of batteries.

Figure 13.   Movements and trajectories to disassemble the cover of batteries.

a Kalman filter is applied employing a noise covariance of the state and measurements equal to 10^{-4} and 10^{-3} respectively. In figure 15, the torques obtained during the experiment, are represented. In this figure the different steps in the disassembly system are represented in two different experiments. This figure represents the torques in x direction (filtered and without applying the Kalman filter) and the filtered forces in the z direction. In figure 15, the

moments in which the robot grasps the cover and in which the cover is rotated to compensate the torques, are shown.

5.4.   Experiment 4. Electronic circuit disassembly from toy

Finally, the fourth experiment consists of disassembling the control electronic circuit of a toy. The electronic circuit is
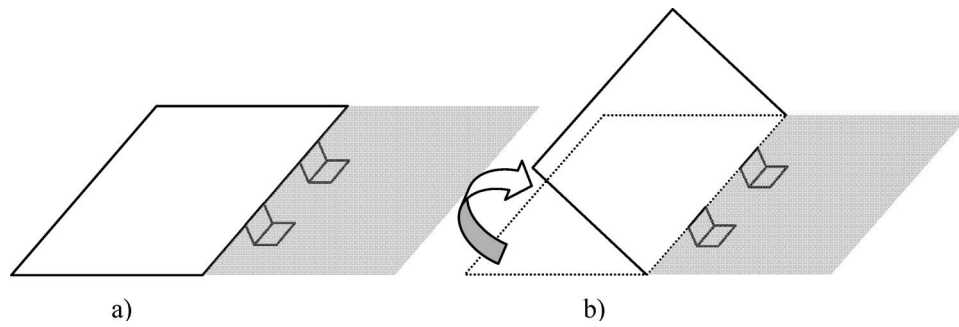
Figure 14. Disassembly movements required for the disassembly of the cover of batteries: (a) grasping; (b) rotation movement.
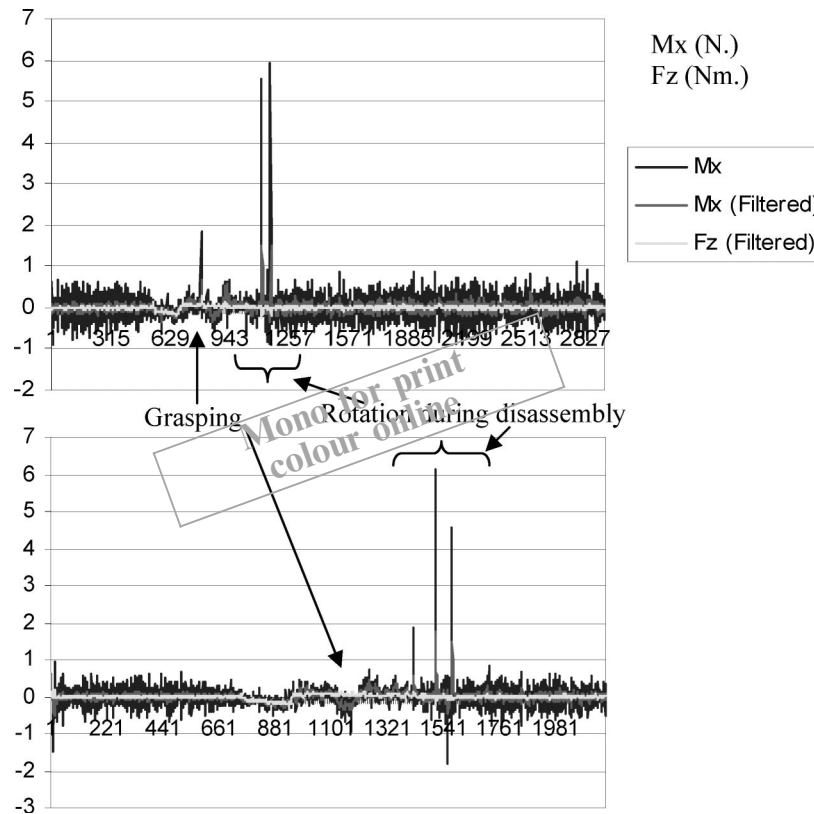

Figure 15. Forces and torques determined from force sensor during the disassembly of the cover of batteries.

joined to the base of the toy with wires and screws. The system must be able to extract the screws which join the circuit to the base of the toy and also it is necessary to cut the wires which join the circuit to different components of the toy. To do these tasks, first,the detection of electronic circuit and its components, board, screws and wires are carried out by means of the strategies showed in section 3. Afterwards, the task planner determines the type of tool that must be used to separate each component. And the vision system combining force information and visual servoing controls the trajectory to reach the target and

the different processes involved in the disassembly task: unscrewing, cover disassembly and cutting wires. It is important to emphasize that in the cutting process of wires, the force sensor is not used; only information obtained from vision is employed.

In figure 16 the steps in the electronic circuit disassembly are shown. First, the detection of the electronic circuit and its screws is shown. Second, the pliers are used to cut the wires that join the circuit to the toy. Third, the task planner determines the exchange of one tool for another, the screwdriver. Thus, the screws are eliminated. Finally, the

Figure 16. Sequence of movements in the disassembly of circuit.

electronic circuit is separated from the base of the toy, then the task planner determines the election of clamps to eliminate the electronic circuit.

6. Conclusions

The necessity to recycle different materials that appear in industrial and electronic components involves defining and developing new systems to disassemble products. In this paper the main components of a new disassembly multi-sensorial system have been presented. The proposed system has been proved to be adequate in disassembly tasks due to different factors. The use of multisensorial systems increases the flexibility of the system. This aspect is very important in recycling products due to the different properties that the product can present once its life cycle is finished. Furthermore, the use of a cooperative robot approach has been proposed. This strategy allows the use of two robots in different situations in which the use of only one robot can not guarantee the correct disassembly. In order to prove the correct behaviour of the system, the results of different experiments are shown which describe the use of the multisensorial systems in real situations.

Nowadays, on the one hand, we are working on improving the robot guidance behaviour during the disassembly task. To do so, new sensorial elements are being included: camera, force sensor, structured light, tactile sensors. The use of high numbers of sensors requires improving the processing of the information in order to guarantee the coherence in the information obtained from each sensor. On the other hand, new strategies and techniques to improve the recognition and detection process of different objects are being studied. Also, strategies to change a view point of camera and reduce the occlusions among the components of an object are being developed. We are also working in the distribution of task not only among robots, but also we are considering a robotic working cell that shares the working area with human operators. This type of approach will give greater flexibility to a flexible automatic disassembly robotic cell.

ensamblado y desensamblado automático (DPI2005-06222)' and by the project 'DESAURO: Desensamblado Automático Selectivo para Reciclado mediante Robots Cooperativos y Sistema Multisensorial' (DPI2002-02103).

References

Bailey, M. and Kuren, V., Flexible robotic demanufacturing using real time tool path Chaumgeneration. *Robot. Comput.-Integr. Mfg*, 2006, **22**, 17–24.

Bouikni, N., Desrochers, A. and Rivest, L., A product feature evolution validation model for engineering change management. *J. Comput. Inf. Sci. Engng*, 2006, **6**, 188–195.

Büker, U., Drüe, S., Götze, N., Hartmann, G., Kalkreuter, B., Stemmer, R. and Trapp, R., Vision-based control of an autonomous disassembly station. *Robot. Auton. Systems*, 2001, **35**, 179–189.

Canny, J., A computational approach to edge detection, *IEEE Trans Pattern Analysis Machine Intell.*, 1986, **8**, 679–698.

Casper, B. and Erik, T., Future disassembly and recycled technology. *Futures*, 1998, **30**, 424–442.

Chaumette, F., Potential problems of stability and convergence in image-based and position-based visual servoing, *The Confluence of Vision and Control*, LNCIS Series, Vol. 237, pp. 66–78, 1998 (Springer Verlag).

Douglas, D. and Peucker, T., Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *The Canadian Cartographer*, 1973, **10**, 112–122.

Eckterh, G., Bönker, T. and Kreis, W., Intelligent disassembly system for industrial reusing and closed loop economy. *Intelligent Assembly and Disassembly*, pp. 105–110, Slovenia, 1998.

Feldmann, K., Trautner, S. and Meeedt, O., Innovative disassembly strategy based on flexible partial destructive tools. *Intelligent Assembly and Disassembly*, Slovenia, pp. 1–6, 1998.

Galambos, C., Kittler, J. and Matas J., Progressive probabilistic Hough transform for line detection. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1999, Vol. 1, pp. 1554.

Gerner, S., Kobeissi, A., David, B., Binder, Z. and Descotes-Genon, B., Integrated approach for disassembly processes generation and recycling evaluation of an end-of-life product. *Int. J. Prod. Res.*, 2005, **43**, 195–222.

Gil, P., Torres, F., Ortiz, F.G. and Reinoso, O., Detection of partial occlusions of assembled components to simplify the disassembly tasks. *Int. J. Advd Mfg Technol.*, 2006, **30**, 530–539.

Güngör, A. and Gupta, S.M., Disassembly sequence plan generation using branch-and-bound algorithm. *Int. J. Prod. Res.*, 2001, **39**, 481–509.

Hershberger, J. and Snoeyink, J., Speeding up the Douglas-Peucker line-simplification algorithm. *Proceedings of 5th Symposium on Data Handling*, 1992, pp. 134–143.

Illingworth, J. and Kittler, J., A survey of the Hough transform. *Comput. Vision, Graphics Image Process*, 1998, **44**, 87–116.

Kuren, M.B.V., Flexible robotic demanufacturing using real time tool path generation. *Robot. Comput.-Integr. Mfg*, 2006, **22**, 17–24.

Manickam, A., Roth, S.D. and Bushman, T., Intelligent and optimal normalized correlation for high-speed pattern matching. *Proc. NEPCON West*, Anaheim, California, 2000.

Matas, J., Galambos, C. and Kittler, J., Progressive Probabilistic Hough Transform, *British Machine Vision Conference*, Southampton, 1998, pp. 256–265.

Mezouar, Y. and Chaumette, F., Path planning for robust image-based control. *IEEE Trans Robot. Automat.*, 2000, **18**, 534–549.

Pomares, J. and Torres, F., Movement-flow based visual servoing and force control fusion for manipulation tasks in unstructured environments. *IEEE Trans Systems, Man, Cybernet. Part C*, 2005, **35**, 4–15.

Pomares, J., Puente, S.T., Torres, F., Candelas, F.A. and Gil, P., Virtual disassembly of products based on geometric models. *Comput. Ind.*, 2004, **55**, 1–14.

Stevels, A. and Rose, C., Environmental metrics to classify end-of-life strategies, *International Symposium on Electronics and Environment*, Denver, 2001, pp. 100–105.

Torres, F., Puente, S.T. and Aracil. R., Disassembly planning based on precedence relations among assemblies. *Int. J. Advd Mfg Technol.*, 2003, **21**, 317–327.

Torres, F., Gil, P., Puente, S.T., Pomares, J. and Aracil, R., Automatic PC disassembly for component recovery. *Int. J. Advd Mfg Technol.*, 2004, **23**, 39–46.

Williams, J.A.S., A review of electronics demanufacturing processes. *Resour. Conserv. Recy.*, 2006, **47**, 195–208.