Research article

Intelligent picking of chaotically stored objects

Rolf Dieter Schraft and Thomas Ledermann

The authors

Rolf Dieter Schraft and **Thomas Ledermann** are based at Fraunhofer Institute for Manufacturing Engineering and Automation IPA, Stuttgart, Germany.

Keywords

Machine vision, Order picking

Abstract

A robot system for the intelligent picking of chaotically stored objects by the analysis of depth data is presented in this research article. This "bin-picking" enables the automatic storing of objects which is a prerequisite for the automatic supply of a processing machine. In order to improve conventional localization systems, special attention is paid – apart from the identification of object positions – to gripping point calculation for the picking and to collision avoidance. To adapt the system to different object geometries, an approach that aims at storing the necessary object data in an off-line generated data base was chosen.

Electronic access

The research register for this journal is available at
http://www.emeraldinsight.com/researchregister

The current issue and full text archive of this journal is available at
http://www.emeraldinsight.com/0144-5154.htm

Assembly Automation
Volume 23 · Number 1 · 2003 · pp. 38–42
© MCB UP Limited · ISSN 0144-5154
DOI 10.1108/01445150310460079

Introduction

"Bin-picking" deals with the localization and picking of chaotically stored objects delivered in a box (Figure 1). Although this is one of the oldest challenges in robotics, up to now, few applicable solutions exist. Only in some cases, and for simple objects, have developers managed to find special solutions and put them into practice. When the principles of mechanical sorting of objects cannot be applied, the objects still are picked and handled manually.

Only a few years ago, the research concerning "bin-picking" concentrated on the analysis of camera pictures (Plate 1). However, due to the rapid increase of the calculating performance in the course of the last few years, especially as far as standard PCs are concerned, and to the development of imaging distance sensors, more complex tasks of object localization can be examined. The advantage of these imaging distance sensors is that the sensor registers the whole scene, measuring the distance to a multitude of surface points of the object. So you get a depth data matrix of the scene surface in consideration, which can be evaluated. For example 3D-laser scanners, 2D-laser scanners with an additional pivoting unit, sheet of light procedures or stereo camera systems are used. In the presented demonstrator cell, a 3D-laser scanner has been used.

Demonstrator cell

The developed robot system continues the research activities for handling chaotically stored objects – an automated robust picking of tires has already been realized by Schraft *et al.* (2000). The extended system is not only able to localize smaller and more complex geometries in one box, but also to avoid collision with the box itself and other parts in the moment of picking. In the realized set-up these are servo-pump casings which are fed to a CNC milling machine.

To guarantee a correct feeding of the machine, after having picked the casings from the box, they are first of all stored temporarily. The objective of doing so is the exact alignment of the casing for its final storage – sometimes they have to be turned around completely. For this, a plane surface was chosen for the temporary storage and a CCD-camera was integrated into the

Figure 1 Total localization process and picking of an object

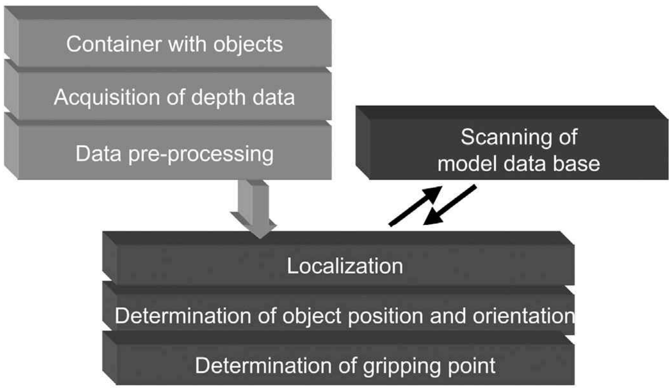
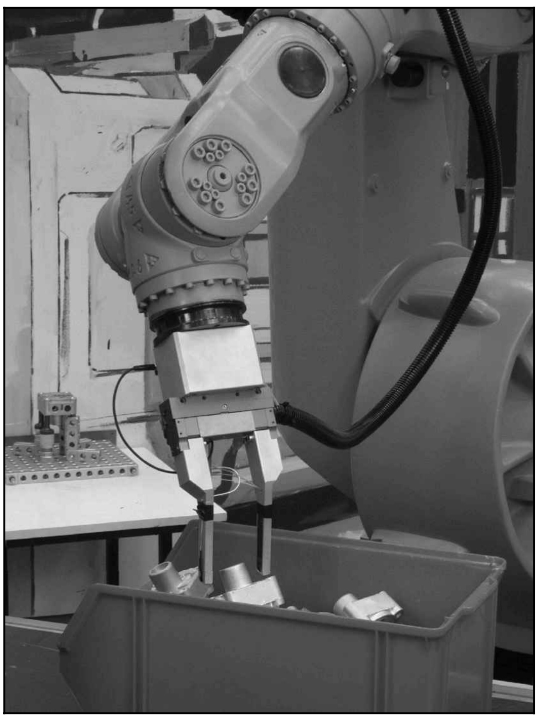


Plate 1 Bin-picking



temporary storage, which determines the position of the object on the surface. Thereby, the requirements for precision on the 3D-localization in the box can be reduced. This is linked with an obvious increase in velocity.

Localization module

The localization module calculates the position and orientation of an object within a box. For an efficient solution, a model-based approach was chosen. This approach uses off-line generated models of the object in different orientations, which are stored in a database. In doing so, only those localization features are determined which can be easily found in the sensor data. By extending the database, the module can be adjusted for different objects. This can be done automatically with a tool, which uses CAD-data to calculate the models in different orientations and stores them in the database. With this tool the user has the possibility to define the accuracy of the database by specifying the increment of rotating the model around every axis. The idea is to generate the models with depth readings equal to the sensor data – increasing the brightness with the distance.

The localization is realized in several steps. First of all the quantity of data is reduced by determining a suitable area of the box. It is then reduced again by fading out pixels which have a depth value lower than a defined threshold. Tests to find an appropriate area of the box showed that the area with the highest object data provides the most important advantage for the following picking operation. The size of the suitable area can be defined by the user to certify that almost one whole object is placed within this area. In our case the side length of the square area was 1.5 times the length of the longest object side.

For the chosen area, the localization is performed on the basis of the off-line generated object models with a normalized gray value correlation. The correlation factor r is calculated by a convolution with the object models as a kernel known from MIL (1999):

$$r = \left[N\sum_{i=1}^{N} I_i M_i - \left(\sum_{i=1}^{N} I_i\right)\sum_{i=1}^{N} M_i\right] \Bigg/ \left\{\left[N\sum_{i=1}^{N} I_i^2 - \left(\sum_{i=1}^{N} I_i\right)^2\right] \times \left[N\sum_{i=1}^{N} M_i^2 - \left(\sum_{i=1}^{N} M_i\right)^2\right]\right\}^{\frac{1}{2}} \quad (1)$$

where N is the number of pixels of the model, I the vector with depth values of the model, and M the vector with depth values of the search area.

Due to the normalization, it is possible to obtain the highest evaluation ($r = 1$) at 100 percent matching and to avoid errors caused by very bright data. As a result of the gray value correlation, the system has information regarding which model from the database fits best to the search area. This data can be directly transferred to the orientation of the object inside the box. Besides this,

Assembly Automation
Volume 23 · Number 1 · 2003 · 38–42

the matching process delivers the position of the object within the sensor data.

To get information about the position for picking the object the following coordinate systems must be defined (Figure 2):

- sensor coordinate system (SKS): the SKS has its origin in the middle of the sensor; the *z*-axis points downwards,
- robot coordinate system (RKS): the RKS has its origin in the foot of the robot; the *z*-axis points upwards,
- box coordinate system (BKS): the BKS has its origin in the left lower corner of the box (seen from above); the *z*-axis points upwards (like the RKS),
- object coordinate system (OKS): the OKS is defined for an object lying on a plane surface with its origin in the middle of the object. It must be transformed in the actual object orientation (Figure 3).

With knowledge of these coordinate systems the position of an object in the RKS can be calculated. Knowing the object position in the SKS the module uses homogenous transformation to calculate the position in the RKS (Figure 4). In the realized systems only the OKS changes due to the position of the object in the box, all other coordinate systems are regarded as static transformations (e.g. the box has a fixed position). The transformation from the RKS to the OKS can be calculated by:

$$T_{\mathrm{RKS}}^{\mathrm{OKS}} = T_{\mathrm{RKS}}^{\mathrm{BKS}} \cdot T_{\mathrm{BKS}}^{\mathrm{OKS}} = T_{\mathrm{RKS}}^{\mathrm{BKS}} \cdot (T_{\mathrm{SKS}}^{\mathrm{BKS}})^{1} \cdot T_{\mathrm{SKS}}^{\mathrm{OKS}} \quad (2)$$

In order to obtain the *z*-coordinate, i.e. the height of the object inside the box, reference points for the models were defined that enable height values to be obtained. The sensor values at these points together with the orientation of the object provide the required *z*-coordinate (Plate 2). As a result, the localization module has received all data for orientation and (*x*, *y*, *z*)-coordinates of the object.

It is to mention that the actual localization module only handles one type of object at the same time. A situation with multiple

Figure 2 Defined coordinate systems within the demonstration cell

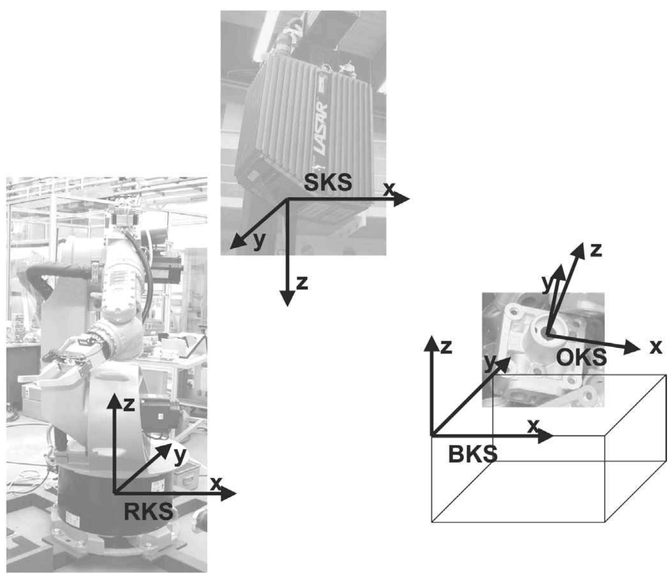


Figure 3 OKS (left: definition, right: transformed)

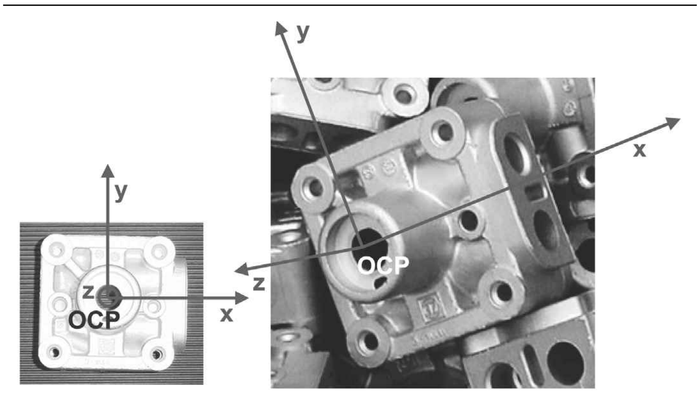


Figure 4 An overview over the used homogenous transformations

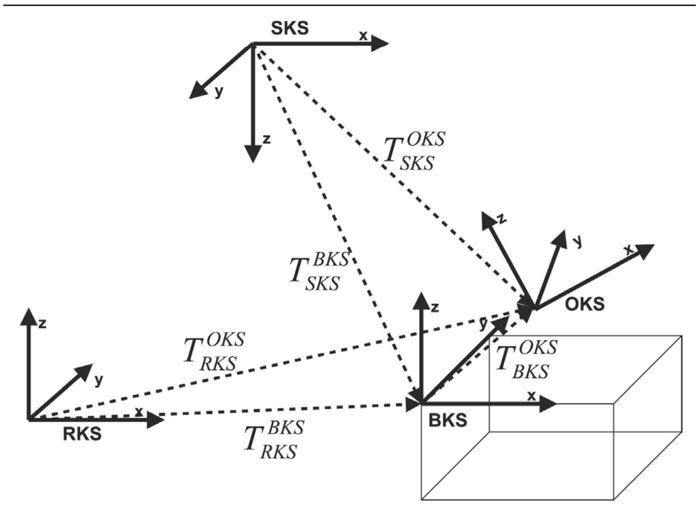
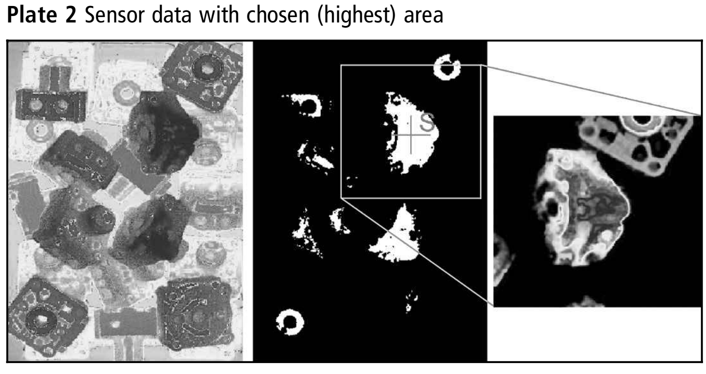


Plate 2 Sensor data with chosen (highest) area



different objects lying within the box is not possible at the moment but can be added in the future.

Calculation of the gripping point and collision avoidance

The module for calculating the gripping point and for collision avoidance is an equally important component of the whole system as the localization module. Only the generation of a suitable gripping point together with the correct approach strategy from the localization data enables the picking and handling of the objects.

For calculating the gripping point, the system gets additional information – taught by the user – about possible gripping points in "zero-position" (rotations around all axes are zero). These points can, after the localization of the object, be rotated into the given object orientation. However, not every rotation of the gripper is useful. Therefore, in addition to every taught grip the degrees of freedom are defined by which the gripper may be rotated for calculating an optimal approach for picking. That means, for every grip, the allowed rotation axes and the maximum permissible rotation angle are determined. On the basis of this information and the restrictions on certain gripper positions, all taught grips for picking can be calculated. The restrictions of non-permissible gripper positions are saved as rotations, respective to the RKS. Gripper positions that are kinematically impossible or not useful, like gripping jaws pointing vertically upwards, can thus be excluded.

To avoid collisions, algorithms known from computer graphics are used. To check collisions of the gripper with the box, an algorithm for volume intersection developed by Foley *et al.* (1990) has been implemented. A collision of the gripping jaws with another object in the box will be detected by using the improved half-line technique developed by Schmitt (1993).

To ascertain the optimal grip, two cases have to be distinguished: if at least one collision-free grip could be found, then the system chooses in accordance with the given rules the best possible grip. If no collision-free grips could be found, an intelligent optimizing algorithm tries to avoid the detected collisions, according to given strategies and considering the gripper's geometries, in order to finally enable the picking. If this process is successful, the best possible grip can be chosen out of all optimized grips. If, in spite of optimization, no collision-free picking is possible, the system asks the user to pick the object manually. Before continuing with the next cycle, the system awaits a confirmation of the manual picking.

The thus found grip will be visualized on a screen for visual checking purposes. Afterwards, the data will be transmitted to the robot control, in order to realize the automated picking of the detected object with the calculated grip.

Results and outlook

The goal of our work was to develop a system for the intelligent picking of chaotically stored servo-pump casings. To guarantee that the system can be adjusted to different objects a model-based approach with an off-line generated database was chosen (Plate 3). The realized demonstrator for automated picking of servo-pump casings shows that the modules for localization, the calculation of the gripping point, and the collision avoidance work reliably. The algorithms for automatically calculating the gripping points and the collisions for any object orientations have proved to be very important for automatic picking (Plate 4). In the case of the used two-jaw gripper with a single degree translation the algorithms were realized.

Experiences with the system showed that due to the camera-support on the temporary storage a localization accuracy of ± 2 mm and 10° for the picking from the box is sufficient.

Plate 3 Some models within the database

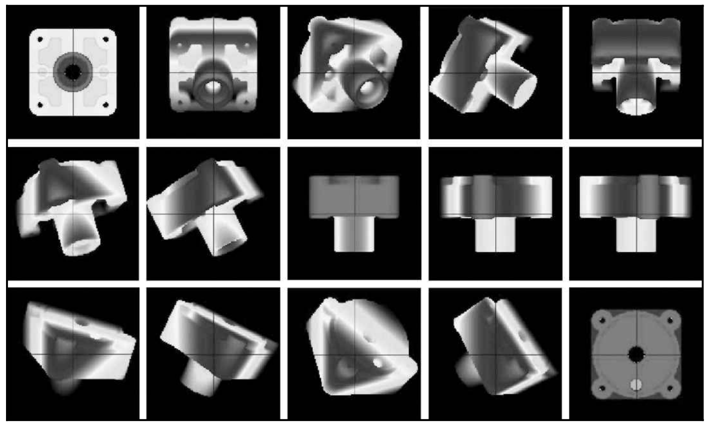
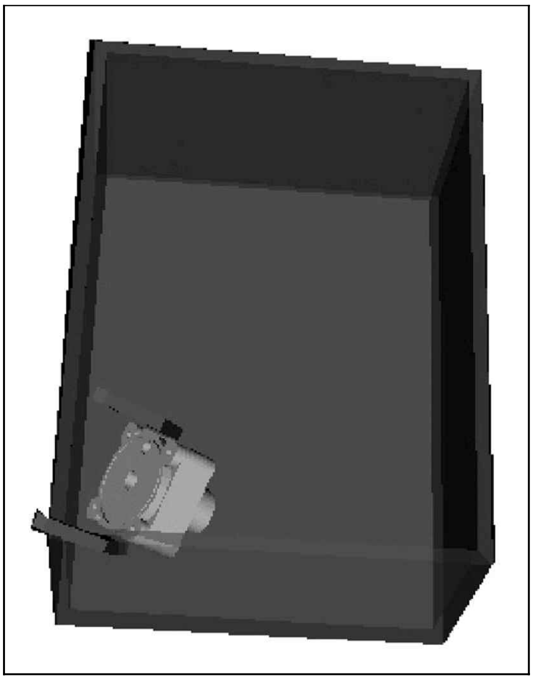


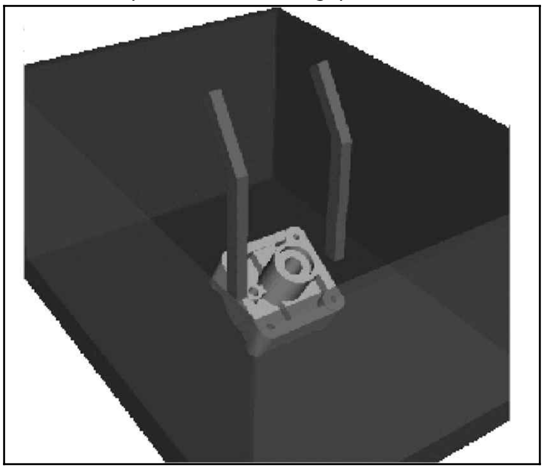
Plate 4 Due to box collision this grip is excluded



The required accuracy for feeding the servo-pump casing to the milling machine is supplied from the camera during the exact alignment. Up to now the system needs about 10 s for an amount of 1,021 × 1,021 pixels to localize one object and to calculate the gripping point (on a standard PC with 1 GHz) (Plate 5).

To adapt the developed system for new object shapes the following steps are required. First the database with the new object models is automatically created with the above-mentioned tool. By adding the information about the size of the suitable area, the localization of one object in the box can be done. For the calculation of the gripping point more intervention from a human being is required. The OKS, the gripping points in "zero-position", the rotation axes of the gripper and the restrictions of gripper positions need to be added manually by the user. This means at the moment high requirements on the knowledge of the user.

Plate 5 Example for a calculated grip



Besides the avoidance of these manual steps for adaptation, a challenge for future improvement are positions inside the box where the systems still needs help from the user. Especially for objects near to an edge of the box the module for gripping point calculation and collision avoidance cannot find a permissible grip although the object is localized correctly. Consequently, future works in the field of intelligent picking will focus on the further development of these algorithms. Especially, the extension to different kinds of gripper geometries and collision areas (e.g. different box geometries), which can easily and intuitionally be added by the user, is necessary for industrial application.

The presented work in research and development were and will be carried out within the project "ParaForm", which is promoted by the German Federal Ministry of Education and Research (BMBF) since 1st October 2001.

References

Foley, J.D., van Dam, A., Feinder, S.K. and Hughes, J.F. (1990), *Computer Graphics: Principles and Practice*, 2nd ed., Addison Wesley, Reading, MA.
MIL (1999), *Matrox Imaging Library Handbuch*.
Schmitt, A. (1993), *Einführung in die graphische Datenverarbeitung*.
Schraft, R.D., Neugebauer, J-G. and Lehmann, K. (2000), "The application of robotics and intelligent systems over, under and around the world", *Depalletizing of Tires with Robust Object Localization*, pp. 206-7.

Further reading

Boothroyd, G. (1992), *Assembly Automation and Product Design*, Marcel Dekker, New York.
Horn, B.K.P. and Katsushi, I. (1984), "The mechanical manipulation of randomly oriented parts", *Scientific American*, Vol. 251 No. 2, pp. 100-11.
Riley, F.J. (1983), *Assembly Automation: A Management Handbook*, Industrial Press, New York.