2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

Combining Visual and Inertial Features for Efficient Grasping and Bin-Picking

Dirk Buchholz, Daniel Kubus, Ingo Weidauer, Alexander Scholz, and Friedrich M. Wahl

Abstract— **Grasping objects is a well-known problem in robotics. If the objects to be grasped are known, usually they are to be placed at a desired position in a desired orientation. Therefore, the object pose w.r.t the gripper has to be known before placing the object. In this paper we propose a simple and efficient, yet robust approach to this challenge, which can (nearly) eliminate dead times of the employed manipulator – hence speeding up the process significantly.**

Our approach is based on the observation that the problem of finding a pose at which the object can be grasped and the problem of computing the pose of the object w.r.t. the gripper can be solved separately at different stages. Special attention is paid to the popular bin-picking problem where this strategy shows its full potential.

To reduce the overall cycle time, we estimate the grasp pose *after* the object has been grasped. Our estimation technique relies on the inertial parameters of the object – instead of visual features – which enables us to easily incorporate pose changes due to grasping. Experiments show that our approach is fast and accurate. Furthermore, it can be implemented easily and adapted to diverse pick and place tasks with arbitrary objects.

I. Introduction and Related Work

Pose estimation of objects is a well-known problem in robotics. Whenever objects have to be manipulated autonomously by a robot, this problem has to be solved. In absence of model data, a very similar problem – referred to as grasping unknown objects – has to be tackled.

The basis for almost all pose estimation techniques is 2D or mostly 3D sensor data. In real world robotic applications, such as industrial work cells or service robots, this data is acquired by 3D sensors, e.g., laser line scanners [1], structured light sensors [2] or the very popular Microsoft Kinect sensor [3].

Two sensor mounting configurations are commonly mentioned in publications related to autonomous grasping. The first solution is to mount the sensor at the end-effector of the robot. This configuration allows for multi-viewpoint measurements but at the cost of time efficiency. Particularly, the scanning procedure cannot be performed concurrently with the manipulation task, which significantly increases cycle times. Furthermore, the sensor itself increases the collision probability and may therefore render found objects ungraspable.

Usually, however, the sensor is stationary and mounted externally. Hence, it can only scan from one direction. To perform a full scan of the entire scene from different directions, several sensors would be necessary or complex hardware and a time-consuming scan procedure would have to be applied.

There are localization approaches that rely on 2D features, e.g. [4]. Here, depth edges are acquired using a multi flash camera mounted at the end-effector, which is very impressive. Unfortunately, the authors do not show real world scenes but isolated objects only. Furthermore, no collision avoidance mechanisms are described.

Most other modern approaches are based on 3D point clouds of the scenes. These point clouds are matched against CAD models ([1], [2], [3], [5]), and then, using a grasp pose relative to the model coordinate system, end-effector poses are determined.

Grasping unknown objects is addressed in [6] for instance. The authors use so called "height accumulated features" that are generated using 3D point clouds. In [7], a similar system is described but the sensor setup is suboptimal, i.e., all grasps are performed into the shadow areas of the scanner. Additionally, the cycle times achieved by both of these systems are quite high.

Another important subtask in autonomous grasping, besides pose estimation of objects, is collision avoidance. Many papers do not address solutions to this problem. There are standard software packages available for this task, e.g. [8], which work on point clouds. Using model-based search algorithms, pick poses are usually computed after the object pose has been determined. If there is no valid grasp for the object, another pose estimation cycle has to be performed or several objects have to be located in the first place. Both options clearly increase cycle times.

Time is the most expensive resource in industry and unsatisfactory cycle times are the main reason why many bin-picking solutions have not found their way into factory production lines yet. Our first strategy to reduce computation times is to analyze the sensor data in an efficient manner. Regarding sensor configurations with a single point of view, the generated 3D points are always well-structured, e.g., they lie on the pixel grid of a Kinect sensor or along a laser line and the moving direction of a linear axis on which the sensor is mounted, etc. Thus, the structure of the range data allows for a representation of the data as 2D images with depth values mapped to intensity values. Representing the data as a 3D point cloud adds a dimensional overhead that does not provide any information gain but increases the computational effort.

When grasping objects, the *object pose* may be determined completely prior to grasping (and used to generate a *gripper*

The authors are with the Technische Universität Braunschweig, Institut für Robotik und Prozessinformatik, 38106 Braunschweig, Germany {d.buchholz, d.kubus, i.weidauer, al.scholz, f.wahl}@tu-bs.de)

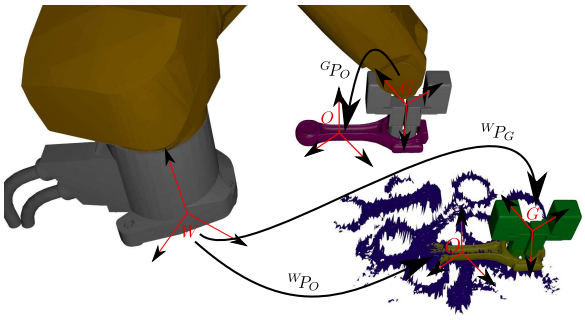


Fig. 1. Definition of important poses for bin-picking. The *object pose* ${}^{W}\boldsymbol{P_O}$ denotes the pose of the object relative to the world, the *gripper pose* ${}^{W}\boldsymbol{P_G}$ describes the pose of the gripper relative to the world to successfully grasp an object and the *grasp pose* ${}^{G}\boldsymbol{P_O}$ denotes the pose of the object relative to the gripper, which is important for defined object placing.

pose relative to it) or merely a valid *gripper pose*, that allows safe grasping of the object, is computed using the scan data. To clarify the meaning of the above mentioned poses, a visual definition cis given in Fig. 1. Regarding the manipulation of known objects, the observation that the object pose w.r.t. the gripper frame (in the following referred to as *grasp pose*) does *not* need to be known *until* the object is actually placed, can be exploited to drastically reduce cycle times.

Rather than estimating the grasp pose solely based on visual clues *before* grasping the object, we grasp the object without determining its pose and use the subsequent transfer motion to obtain a pose estimate based on its inertial parameters. As less computation time is required *before* grasping and the pose estimate is computed *during* the transfer motion – which has to be executed anyway – a significant cycle time reduction can be achieved. A more subtle but valuable advantage is that the pose estimate is not deteriorated if the object moves when grasping it. Based on the above observation we have devised and implemented an efficient solution to the bin-picking problem of both known and unknown objects.

Besides presenting a very efficient approach to generate valid pick poses in Section II, we show that it is very important to understand the sensor data as what it really is – a depth *image* – which hence allows to reduce the computational complexity from 3D to 2D. In Section III we review and extend an unconventional but very effective technique to grasp pose estimation based on inertial parameters – which allows our approach to be applied to bin-picking of known objects. The proposed approaches have been evaluated experimentally with both known and unknown objects as presented in Section IV. In Section V we conclude our work and give an outlook on open questions.

II. Fast Gripper Pose Estimation

Understanding the environment is at the heart of any autonomous robotic system. In the context of robotic bin-picking systems, this predominantly means the analysis of 3D sensor data for gripper pose estimation. A gripper pose is a pose ${}^{W}\boldsymbol{P_G}$ of the gripper w.r.t. the world frame $\boldsymbol{W}$ at which an object can be grasped and is thus sufficient to actually grasp the object. However, many bin-picking approaches determine gripper poses using 6D object poses that are computed in advance. This method has the advantage that the complete pose of the object is known and thus the object can be placed in a defined manner. The disadvantage of this indirect gripper pose estimation is that situations may occur in which objects have been localized but no valid, collision-free gripper poses can be found. In that case, further objects have to be localized, which results in time overhead.

To avoid this problem, we propose a very simple and efficient direct gripper pose estimation technique. A significant advantage of direct gripper pose estimation is that by definition no invalid gripper poses can occur. Since no model data is available, unknown objects cannot be placed at a well-defined pose but only at a desired position. The following subsections describe our approach to grasping unknown objects.

A. Data Interpretation

To keep the system simple and effective, we use the Kinect sensor as depth sensor in our experiments. The Kinect sensor can generate depth data at a high rate using a single camera shot, i.e., in one thirtieth of a second. The short scan time comes at the cost of very noisy data and a low resolution of the depth map. If scan time is not an extremely critical issue, n images of the Kinect can be averaged, reducing noise at the cost of a longer scan time. To overcome thermal and drift issues, as described in [9], an online calibration technique is used that corrects depth values using three known markers in the work space. The known world coordinates of these markers ($\vec{\rho}_i, i = 1..3$, see Fig. 7) and the measured coordinates in the depth camera image are used to correct the calculated gripper poses.

As already mentioned, in industrial setups, sensors are commonly mounted at a suitable location in the work cell and have a fixed view point. 3D data generated in this fashion is always well-structured, which means that the 3D points lie on a grid and that there are no points occluding each other when using the viewing direction of the scanner. Treating these data as a 3D point cloud is therefore suboptimal. The data should rather be interpreted as 2D images which results in a dimension reduction without any loss of information.

As the sensor is subject to occlusion and shadowing effects, it is advisable to interact with the scene using the viewing direction as approach direction for the robot end-effector. In this way, no collisions can occur due to occlusions. Since two degrees of freedom (DoF) of the end-effector pose are therefore defined by the sensor setup, the gripper pose estimation can be completely computed in 2D.

B. Generation of Gripper Pose Hypotheses Using Matched Filters

Noisy gray value images with low resolution were already available 30 years ago. This is about the time the bin-picking problem was first addressed. Due to this relation – but with the difference that the noisy images contain depth values nowadays – classic approaches can constitute the basis for

modern approaches to this problem. In [10] an approach for gripper pose estimation using matched filters is presented. Amazingly, there have not been further developments of these ideas. We apply these ideas to modern depth images and acquire a very efficient gripper pose estimation technique.

A gripper pose is a feature in the depth image that shows a local decrease in depth. In other words, a pattern must be found that matches the gripper footprint in its appearance. The gripper used in the experiments is a standard parallel jaw gripper. As the gripper may be rotated around its approach vector and the algorithm shall be as fast as possible, we generate a filter kernel that is rotationally symmetric and thus contains all possible gripper orientations (see Fig. 3(a)). The correlation C of the depth image I_D and this kernel K_G yields all possible gripper poses as these are the local maxima of the correlation function. To obtain correct results, the perspective characteristic of the Kinect depth map has to be considered in the correlation. Pixels in the image are larger if they are farther away from the camera. So, a scaling factor s is computed as the approximate pixel size of the camera image at the topmost surface point. s can be found using the intercept theorem and the intrinsic parameters of the depth camera. The kernel K_G can then be resized by this factor and a scaled filter kernel $K'_G = s \cdot K_G$ is obtained. The locations (x_m, y_m) of the maxima of $C(x, y)$ define two DoFs (x, y) of the gripper pose in pixel coordinates (an example can be found in Fig. 2). These coordinates then have to be transformed into the sensor coordinate system. The

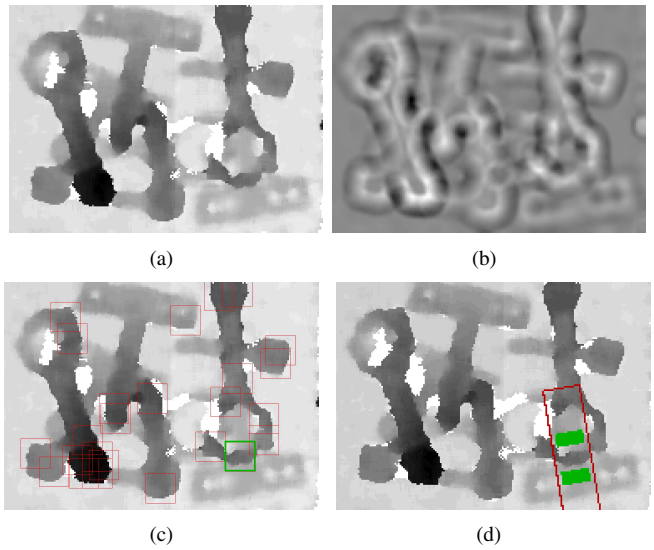


Fig. 2. Gripper pose estimation by correlation. (a) Depth image of an analyzed pile of different objects. Dark pixels are closer to the camera. (b) Correlation result C. Regions that are high and have low surroundings produce the highest (brightest) results. (c) Generated hypotheses using local maxima of C. The best maximum (green square) is further analyzed in Fig. 3. (d) Valid gripper pose shown as superimposed gripper footprint.

third DoF can be found by the local depth $z = I_D(x_m, y_m)$ in the image or in an average of n successive images.
The last DoF, i.e., the rotation angle around the approach vector, has to be computed using a local patch of the image around the maximum. A threshold is applied to the patch, sized like the filter kernel, to generate a binary image in which the maximum values are set to 1 and the minimum values to 0. Then, the topological skeleton of the binary patch is computed. Using only the skeleton, a line is fitted into the patch using RANSAC [11]. (Alternatively deterministic linear regression techniques could be applied. The number of skeleton pixels is around $20 - 30$, depending on the scaling factor s, which results in a very fast evaluation.) The normal on the estimated line describes the optimal orientation Φ of the gripper (see Fig. 3). Now, only one additional parameter,

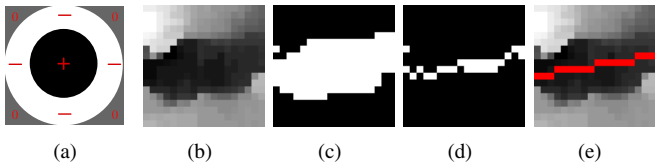


Fig. 3. Estimation of the gripper orientation Φ of the hypothesis marked green in Fig. 2(c). (a) Filter kernel used to generate the hypotheses. (b) Section around the pose hypothesis (x_m, y_m). (c) Binary image of the section. (d) Topological skeleton of the section. (e) Estimated line using RANSAC.

the approach depth d_a, has to be found. This parameter contains information about how far the gripper may approach the gripper pose and can easily be estimated using the pixels at the computed jaw positions. In this context, the jaw position means the footprint hull of the jaws during their close or open motion. Subtracting z from the minimum depth of all pixels covered by the gripper footprint yields d_a. Fig. 4 illustrates its computation.

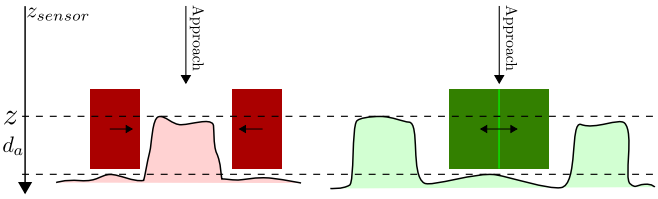


Fig. 4. Calculation of the approach distance d_a using outer grasps or inner grasps.

If the objects to be grasped contain regions that allow for a grasp from their inside, the local minima of the correlation function can be used as gripper pose hypotheses.

C. Collision Avoidance and Gripper Pose Computation

With the algorithm described above, many hypotheses are generated. The best hypothesis has to be found in a second step. One quality measure is the parameter d_a, which is already available, and assures that no collisions occur between the gripper jaws and the scene. Additionally, it has to be checked that d_a exceeds a certain threshold ϵ_p and that the gripper palm does not collide with any obstacle. The threshold ϵ_p has to be set to fit the gripper properties. By analyzing the palm footprint, which is a simple rectangle in the depth map, in the same manner as d_a has been obtained, full collision avoidance can be guaranteed. The described

steps to compute a valid gripper pose are summarized in Algorithm 1. Note that the output ${}^{\mathbf{W}}\mathbf{P}_{\mathbf{G}}$ does not contain the pitch and yaw angles since they are fixed by the sensor setup as described above.

Data: depth image I_D, gripper-defined kernel K_G
Result: Gripper Pose ${}^{\boldsymbol{W}}\boldsymbol{P}_{\boldsymbol{G}} = \{x_g, y_g, z_g, \Phi\}, d_a$
find global minimum M_d in I_D;
scale K_G according to M_d;
Correlation function $C = I_D \star\star K'_G$;
while *no valid pose found* **do**
  find global maximum M in C;
  $x_m \leftarrow M.x$;
  $y_m \leftarrow M.y$;
  $z_g \leftarrow I_D[x_m, y_m]$
  estimate topological skeleton line l in patch around (x_m, y_m);
  $\Phi \leftarrow l.normal$;
  estimate approach depth d_a;
  if $d_a > \epsilon_p$ **then**
    if *gripper palm is collision-free* **then**
      valid pose found;
    else
      continue;
    end
  else
    continue;
  end
  delete local area in C around (x_m, y_m);
end
transform (x_m, y_m) from pixel values into world coordinates;
$x_g \leftarrow T(x_m)$;
$y_g \leftarrow T(y_m)$;

Algorithm 1: Gripper pose estimation.

D. Limitations and Enhancements

Parallel jaw grippers are very common in industry but, of course, other types of grippers are widely used as well. If grippers with significantly different geometries are employed, e.g., 3-jaw grippers or anthropomorphic hands, the collision avoidance step has to be adapted to the respective gripper footprint while the preceding processing steps essentially remain unchanged.

Since the object pose is not determined prior to grasping, unfavorable grasp poses may considerably restrict the set of feasible place poses – depending on the object, gripper, and fixture geometry as well as the characteristics of the employed manipulator.

As described above, using the viewing direction of the sensor as the approach direction has significant advantages. But, it may also be beneficial to modify the approach direction by computing suitable pitch and yaw angles. An analysis of the local patch and an estimation of the principal components of the segmented regions can be used to calculate these angles. The collision avoidance mechanism has to be adapted in this case as the projection of the gripper into the depth image changes. To tackle this problem, for instance the approach proposed in [1] can be used.

III. Fast Bin-Picking Exploiting Inertial Object Features

So far, unknown objects may be grasped from a pile or a bin and may be transfered to a predefined *position*. For industrial applications or service robotics, it is often essential to place the grasped objects at a predefined *position* in a desired *orientation*. In these cases, model data of the objects is required.

Employing the approach to gripper pose estimation approach presented in Section II has the drawback that the pose of the object w.r.t. the gripper (denoted as grasp pose) is unknown. Regarded from a different point of view, it has the essential advantage that only the parameters *needed* for grasping are determined *prior to* grasping. Thus, the grasp motion can be initiated earlier; dead times of the robot can be reduced and therefore pick and place cycle times can be minimized.

Nevertheless, to be able to place the grasped object at a desired pose, the grasp pose has to be determined *before* the object is *placed*. In this section, an approach is described that estimates the grasp pose of the object *during* the transfer motion *after* the object has been grasped. In contrast to established techniques, our approach mainly relies on the inertial parameters of the object instead of visual clues.

In fact, by estimating the coordinates of the center of mass of the object and its principal axes of inertia, a finite set[1] of pose hypotheses can be obtained.

The first subsection reviews a procedure to estimate the inertial parameters of an object that has been grasped. In the following subsection, the robust derivation of pose hypotheses from inertial parameters is addressed and strategies to deal with pose ambiguities are proposed. The last subsection demonstrates how the pose estimation approach is embedded in our bin-picking system.

A. Online Inertial Parameter Estimation

The online estimation of inertial parameters has already been proposed in [12] and its accuracy has been improved in [13]. Therefore, the approach is merely reviewed briefly for the sake of completeness.

The inertial parameters of an object are its mass m, the coordinates of the center of mass (COM) ${}^{R}c_x, {}^{R}c_y, {}^{R}c_z$ w.r.t. a given reference frame $\boldsymbol{R}$ and the elements of the inertia matrix $\boldsymbol{I}$, cf Eq. (1).

$$\boldsymbol{I} = \begin{pmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{xy} & I_{yy} & I_{yz} \\ I_{xz} & I_{yz} & I_{zz} \end{pmatrix} \tag{1}$$

Its elements are the moments and products of inertia. The ten inertial parameters can be compiled into a vector ${}^{S}\boldsymbol{\varphi}$.

[1] not considering ambiguities arising from symmetry

$$
\begin{aligned}
{}^{S}\boldsymbol{\varphi} \quad = \quad & \big[m, m\,{}^{S}c_x, m\,{}^{S}c_y, m\,{}^{S}c_z, \\
& {}^{S}I_{xx}, {}^{S}I_{xy}, {}^{S}I_{xz}, {}^{S}I_{yy}, {}^{S}I_{yz}, {}^{S}I_{zz}\big]^T
\end{aligned} \tag{2}
$$

The superscript $\boldsymbol{S}$ indicates that the sensor frame is used as the reference frame here. Note that the coordinates of the center of mass have been multiplied by the mass to obtain a linear parameter estimation problem. Hence, the problem of estimating the inertial parameters of a load attached to a manipulator can be cast into matrix form:

$$
\begin{pmatrix} {}^{S}\boldsymbol{f} \\ {}^{S}\boldsymbol{\tau} \end{pmatrix} = \boldsymbol{V}\left({}^{S}\boldsymbol{a}, {}^{S}\boldsymbol{g}, {}^{S}\boldsymbol{\omega}, {}^{S}\boldsymbol{\alpha}\right){}^{S}\boldsymbol{\varphi} \tag{3}
$$

where ${}^{S}\boldsymbol{f}$ and ${}^{S}\boldsymbol{\tau}$ denote measured forces and torques. The 6×10 data matrix $\boldsymbol{V}$ depends on the linear acceleration ${}^{S}\boldsymbol{a}$, gravity ${}^{S}\boldsymbol{g}$, angular velocity ${}^{S}\boldsymbol{\omega}$, and angular acceleration ${}^{S}\boldsymbol{\alpha}$. In contrast to established approaches (e.g. [14], [15]), the inertial parameters are to be estimated *during* the transfer motion. Therefore, recursive estimation techniques have to be employed. We apply a weighted recursive instrumental variables technique ([12], [16]), which combines signals of several sensors.

A wrist-mounted inertial measurement unit provides angular velocity and linear acceleration signals. A wrist-mounted force-torque sensor measures forces and torques as well as linear and angular accelerations; the encoder signals of the manipulator are Kalman filtered as proposed in [12] and serve as a secondary source for angular velocity, linear acceleration, and angular acceleration.

Since the force-torque sensor measures the forces-torques exerted by the grasped object as well as those exerted by the gripper, the forces and torques resulting from the gripper have to be removed from the measurements or alternatively the inertial parameter vector of the gripper ${}^{S}\boldsymbol{\varphi_{gripper}}$ has to be subtracted from the estimated parameter vector ${}^{S}\hat{\boldsymbol{\varphi}}_{\boldsymbol{total}}$ to yield an estimate of the inertial parameters of the object ${}^{S}\hat{\boldsymbol{\varphi}}_{\boldsymbol{obj}}$.

$$
{}^{S}\hat{\boldsymbol{\varphi}}_{\boldsymbol{obj}} = {}^{S}\hat{\boldsymbol{\varphi}}_{\boldsymbol{total}} - {}^{S}\boldsymbol{\varphi}_{\boldsymbol{gripper}} \tag{4}
$$

Note that practical challenges, such as the elimination of sensor offsets, etc., have to be addressed to obtain robust and reliable estimates. A detailed discussion of extensions to the sketched approach can be found in [13].

B. Online Pose Estimation

Based on the estimated inertial parameters of the object, estimates of the coordinates of the center of mass (COM) as well as the principal axes of inertia can be obtained according to [14], [12]. To compute the principal axes of inertia, the inertia matrix ${}^{S}\boldsymbol{I}_{\boldsymbol{obj}}$ is first expressed w.r.t. the COM using the parallel-axis theorem (cf e.g. [17]). The eigenvectors of the resulting matrix ${}^{COM}\boldsymbol{I}_{\boldsymbol{obj}}$ constitute the principal axes of inertia. The eigendecomposition of ${}^{COM}\boldsymbol{I}_{\boldsymbol{obj}}$ is given by Eq. (5).

$$
{}^{COM}\boldsymbol{I}_{\boldsymbol{obj}} = {}^{S}\boldsymbol{R}\,\boldsymbol{I_p}\,{}^{S}\boldsymbol{R}^{T} \tag{5}
$$

The columns of ${}^{S}\boldsymbol{R}$ are the eigenvectors and $\boldsymbol{I_p}$ is a diagonal 3×3-matrix containing the eigenvalues which constitute the principal moments of inertia; in the following $\boldsymbol{I_p}$ will be used as an indicator of the estimation error. The matrix of eigenvectors can be interpreted as a rotation matrix ${}^{S}\boldsymbol{R}$ relating the orientation of the principal axes to the sensor frame $\boldsymbol{S}$. Therefore, a pose hypothesis ${}^{S}\boldsymbol{P}$ can be composed of ${}^{S}\boldsymbol{R}$ and a translation matrix given by ${}^{S}\boldsymbol{T} = Trans\left({}^{S}c_x, {}^{S}c_y, {}^{S}c_z\right)$.

$$
{}^{S}\boldsymbol{P} = {}^{S}\boldsymbol{T}\,{}^{S}\boldsymbol{R}\,\boldsymbol{M} \tag{6}
$$

Here, the matrix $\boldsymbol{M}$ may denote the identity matrix $\boldsymbol{E}$ or involve any of the following rotations: $Rot\,(x, \pi)$, $Rot\,(y, \pi)$, and $Rot\,(z, \pi)$.

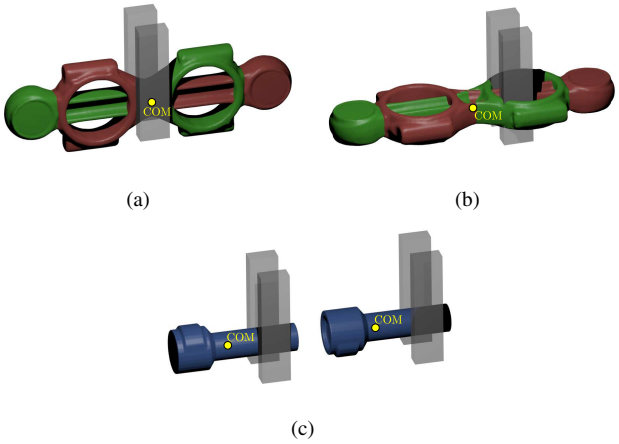
This pose ambiguity results from the ambiguity of the principal axes of inertia. It may be tolerated in object recognition [12] and related applications but certainly not in bin-picking – where placing a grasped object in a specific pose is a key requirement. The pose ambiguity can be resolved by incorporating geometric knowledge of the object and the gripper and/or additional information from visual features. Note that these features can be computed *during* the transfer motion and thus these computations do generally not affect the cycle time. As a matter of fact, alternative hypotheses can often be excluded easily since the jaw geometry of the gripper restricts one or more DoFs of the object and the knowledge of the jaw distance or at least the minimum and maximum values can reduce the set of possible poses further. Thus, commonly only one pose hypothesis remains.

So, for most objects, the set of pose hypotheses will collapse to a single one. However, the number of potential pose hypotheses may also increase in case of nearly-symmetric objects as the parameter estimates and hence the pose estimates will be affected by errors that prevent the exclusion of alternative hypotheses and may also result in additional hypotheses. But even in these cases, the extraction of additional features from the depth image can effectively eliminate alternative hypotheses as the hypotheses typically differ considerably. Fig. 5 depicts different scenarios illustrating the mentioned ambiguities as well as approaches to resolve them.

C. Inertial Pose Estimation for Bin-Picking

To minimize the error in the estimates, the parameter estimation process is usually based on dedicated excitation trajectories, e.g., sinusoidal trajectories that are parameterized such as to minimize the condition number of the data matrix [12]. As our goal is to minimize cycle time, we modify standard transfer trajectories by superposing sinusoidal functions in the three hand joints. The resulting moderate increase in the condition number does not influence estimation errors notably.

Since the modified transfer trajectories are not time-optimal, the cycle time can be further reduced by switching to a (nearly) time-optimal place trajectory as soon as the



(a) (b) (c)

Fig. 5. Ambiguities and ambiguity elimination. (a) The two depicted poses of the piston rod cannot be distinguished based on the inertial features. Simple clues provided by the vision system, however, can easily resolve the ambiguity as the pose hypotheses differ considerably. (b) The pose hypothesis associated with the green piston rod can be discarded. The gripper geometry and the maximum jaw distance render the depicted grasp impossible. (c) The pose hypotheses cannot be distinguished *in practice* because the object is nearly rotationally symmetric and therefore even minor estimation errors will lead to significant errors in the computation of the principal axes. However, image features may resolve such ambiguities as well.

parameter uncertainty drops below a tolerable threshold. Usually, the parameter covariance is used to evaluate the uncertainty in the estimates. As ground truth is available, i.e., the inertial parameters of the objects are known (or can at least be estimated accurately employing dedicated estimation trajectories), switching between the trajectories is decided based on the symmetric Kullback-Leibler divergence (SKLD) [18] of two invariant feature vectors. One vector is obtained from the inertial parameters of the object estimated during the transfer motion and the other vector is derived from ground truth data.

Regarding the parameter vector ${}^{S}\boldsymbol{\varphi}$, merely the mass m is invariant under translation and rotation of the object w.r.t the gripper. However, the principal moments of inertia, i.e., the diagonal elements of $\boldsymbol{I_p} = diag\,(I_1, I_2, I_3)$, yield three additional features which are also invariant w.r.t. translation and rotation. Hence, four invariant features can be compiled in a feature vector $\boldsymbol{f} = [m, I_1, I_2, I_3]^T$.
In layman's terms, the SKLD can be regarded as the "distance" between two probability distributions. To compute this "distance", the covariance has to be tracked when computing the feature vector based on the estimated parameter vector ${}^{S}\hat{\boldsymbol{\varphi}}$ and its covariance matrix $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\varphi}}}$. The scaled unscented transform [19] is utilized for this purpose. Thus, a feature vector $\boldsymbol{f_O}$ and a corresponding covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{f_O}}$ are obtained from the estimated inertial parameters. The required reference feature vector $\boldsymbol{f_Q}$ can be obtained from CAD data. Assuming multivariate Gaussian probability density functions, the SKLD J_{KL} of the feature vectors $\boldsymbol{f_O}$ and $\boldsymbol{f_Q}$ and their covariance matrices can be computed by Eq. (7) where $\boldsymbol{\Delta f} = \boldsymbol{f_O} - \boldsymbol{f_Q}$.

$$J_{KL} = \frac{1}{2}\Big[\boldsymbol{\Delta f}^T\left(\boldsymbol{\Sigma}_{\boldsymbol{f_Q}}^{-1} + \boldsymbol{\Sigma}_{\boldsymbol{f_O}}^{-1}\right)\boldsymbol{\Delta f} + tr\left(\boldsymbol{\Sigma}_{\boldsymbol{f_Q}}^{-1}\boldsymbol{\Sigma}_{\boldsymbol{f_O}} + \boldsymbol{\Sigma}_{\boldsymbol{f_O}}^{-1}\boldsymbol{\Sigma}_{\boldsymbol{f_Q}} - 2\boldsymbol{E}\right)\Big] \quad (7)$$

tr denotes the trace of a matrix. The switch from the estimation trajectory to the place trajectory occurs when J_{KL} falls below a predetermined threshold.

To enable instantaneous switching from the estimation trajectory to a direct place trajectory, an online trajectory generator as well as a suitable robot programming paradigm are essential. We employ a jerk-limited online trajectory generator [20] and an extension of the so-called manipulation primitive paradigm [21] which allows for instantaneous (within one control cycle, i.e., 2ms) switching of target poses and target velocities.

Fig. 6 illustrates the processing cycle of the discussed bin-picking system. First, an image of the bin is acquired and a gripper pose ${}^{W}\boldsymbol{P_G}$ is generated. Then, the object is grasped while additional visual features *may* be computed if necessary. Subsequently, the estimation trajectory is executed and another image may be acquired as soon as the manipulator is out of the sensor view. Immediately after the start of the estimation trajectory the inertial parameters of the object are estimated and the inertial features as well as the KL divergence w.r.t. a reference feature vector are computed in *each* time-step. If the KL divergence exceeds a predetermined threshold, another estimation step is executed. Otherwise, pose hypotheses are generated and ambiguities are resolved as explained earlier. Subsequently, the pose hypothesis is used to specify the place pose which is then approached. A new gripper pose has already been computed when the robot reaches the place pose and the cycle is repeated.

IV. Experimental Setup and Results

To evaluate the proposed approaches, a Stäubli RX60 robot with an open control architecture [22] and a Microsoft Kinect sensor have been used (cf Fig. 7). A wrist-mounted sensor manufactured by JR3 (85M35A3-I40-D 200N12) provides 6D force-torque as well as 6D acceleration measurements. Angular velocity measurements are supplied by an Analog Devices IMU (ADIS 16364). For vision analysis, a $3.6GHz$ Windows PC with $8GB$ RAM is used. The inertial parameter estimation is performed on a PC running the QNX Neutrino real-time operating system. A distributed real-time middleware [23] enables efficient communication between the nodes. All components are situated in a robotic work cell with a known world frame. To avoid collisions between the robot and the sensor, the Kinect sensor is mounted $109cm$ above the bin. A standard pneumatic parallel jaw gripper is used as these grippers are rugged and therefore common in industry. Two series of experiments, as described in the following two subsections, are performed. In the first series, the ability of the system to grasp unknown objects is investigated. For this purpose, several different objects are picked and placed in a bin. The second series serves

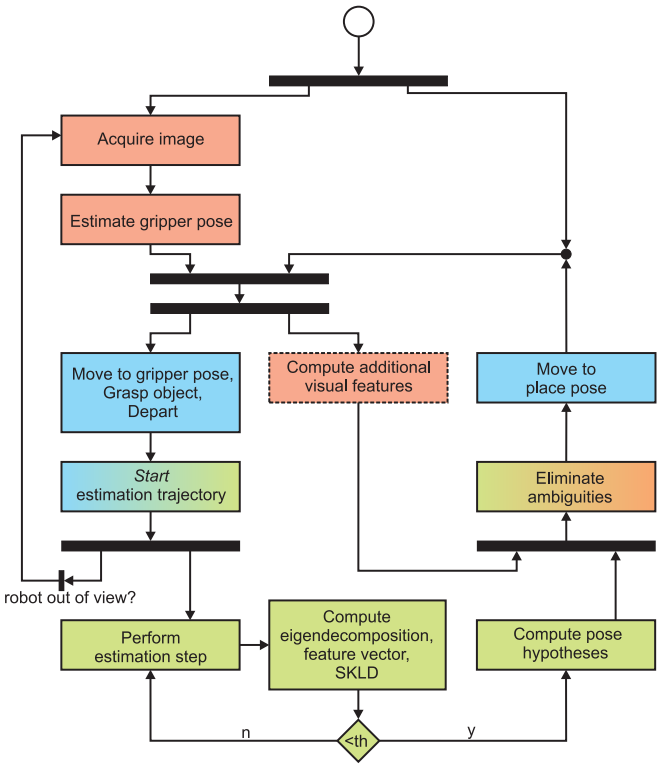


Fig. 6. Simplified diagram of the bin-picking process.

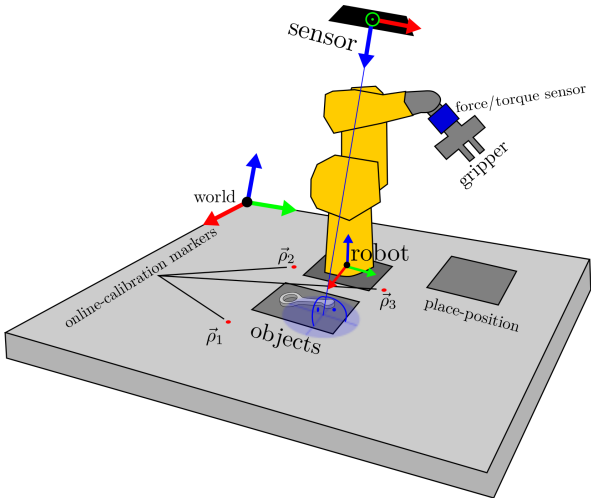


Fig. 7. Experimental setup.

to evaluate the applicability of the system as a bin-picking solution. With the aim of emulating industrial production environments, industrial metal parts in a bin are grasped and placed at a desired pose. In the first series, outer grasps are used whereas in the second series, inner grasps are employed.

A. Grasping Unknown Objects

To analyze the performance of the system with unknown objects, we prepare several piles of objects in the work space (examples can be seen in Fig. 8). As test objects, raw metal parts as well as plastic and wooden objects of arbitrary shapes are used. The piles have to be cleared into a bin. A

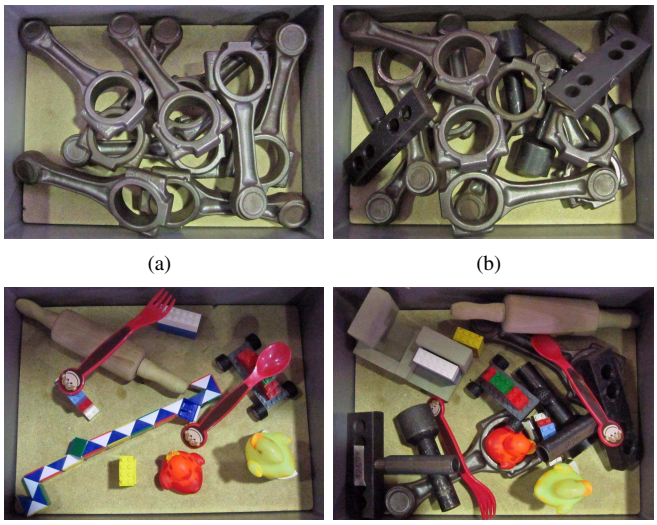


Fig. 8. Four example test piles used in the experiments. The bin is filled with (a) identical metal parts, (b) mixed metal parts, (c) plastic toys and wooden toys, and (d) a mix of all available objects.

series of 261 grasp attempts has been performed. In contrast to the experiments presented in the following subsection, force sensing is merely employed to detect collisions. The data acquisition time is $n/30$ seconds where n is the number of averaged Kinect scans. In our experiments, n is set to 3. The vision processing time, i.e., the time until a valid gripper pose becomes available, is $18ms$ (no notable variance). This value is comprised of $10ms$ for convolution and maxima search as well as $8ms$ for maxima evaluation including the computation of the approach distance and the orientation of the gripper. Obviously, both image acquisition and vision processing (in total requiring $318ms$) can be executed during the place movement of the manipulator. With our test setup, the overall cycle time is 8 seconds from pick to pick, which is exactly the duration of the robot motion. Note that the dynamics of the robot are reduced significantly since a pneumatic load limiter is used to prevent damages of the manipulator in case of collisions. However, this load limiter is also triggered by inertial forces and associated torques.
The total grasp success rate is 95.4%. Unsuccessful grasps are caused by sensor noise and may be classified into two categories. The computed approach distance may be too small to allow for a stable grasp. In this case, the objects slip off the gripper during the depart motion. Sensor noise may also result in minor collisions of the gripper with the object thus compromising the grasp. Due to the short acquisition and vision processing times, a new gripper pose can be computed without significant time overhead in these rare cases to immediately retry grasping. The average cycle time is therefore not increased notably.

B. Bin-Picking

To analyze the applicability of the system to the bin-picking problem, a bin filled with known objects (piston rods, cf Fig. 8(a)) is used. The objects are grasped using an

inside grasp (using the minima of the correlation function as gripper pose hypotheses) and placed into a device emulating a machine feeding system. During the transfer motion the procedure of Section III is applied to estimate the grasp pose and to adapt the place motion. Due to the already mentioned dynamics restrictions, the duration of the transfer motion (from the bin to the feeder) increases by approx. 20% if sinusoidal functions in the hand joints are superposed to enable the estimation of inertial parameters. However, as the estimation trajectory is canceled as soon as a sufficient pose estimate is available, the effective increase in the duration of the transfer motion is lower. As the inside grasp eliminates all but two DoFs of the object, which can be estimated unambiguously, no additional visual features are required. Note that the pose ambiguity due to the symmetry of the object is not considered here. The key parameter is the angle of the piston rod around the approach vector of the gripper. The average absolute error obtained in 50 trials is $2.7°$. This error enables a safe placement in the feeder as it can compensate minor angular deviations. A significant fraction of the estimation error is due to structural oscillations of the manipulator during the estimation trajectory. This problem is further aggravated by the load limiter.

V. Conclusion and Outlook

In this paper we have proposed a very simple and efficient approach to autonomous grasping of objects. We have demonstrated that it is advantageous to treat 3D sensor data as images rather than 3D point clouds. In this way, vision analysis times can be reduced drastically if single vision sensor (and single view point) setups are used. Our algorithms are very robust against noise. So, single shot depth cameras (e.g., the Kinect sensor) can be employed, which reduces acquisition and analysis times even further.

We have extended our approach to bin-picking tasks by exploiting the observation that the grasp pose of the object need not be known until it is actually placed. As a consequence, we estimate the grasp pose between grasping and placing the object, viz. during the transfer motion. The estimation is based on the inertial parameters of the object and can be supplemented by additional visual features if necessary.

Our approach reduces the cycle time of bin-picking and autonomous grasping tasks nearly to the time required to manipulate the object. Thus, the bottleneck of bin-picking applications is shifted from visual data processing to the dynamics of the manipulator. We outperform many state-of-the-art approaches w.r.t. cycle times although the maximum acceleration of the manipulator is restricted due to the employed safety devices – particularly the load limiter. Our system can be re-implemented with low-cost vision hardware and comparatively low effort in software development.

Future work will aim at embedding the described sensor fusion approach into a thorough probabilistic framework. Moreover, we will develop approaches to solve the mentioned ambiguity problem in a structured and generic manner. To mitigate the disturbances due to structural oscillations, the RX series manipulator will be replaced by a TX series manipulator which shows more favorable characteristics.

References

[1] D. Buchholz, M. Futterlieb, S. Winkelbach, and F. M. Wahl, "Efficient bin-picking and grasp planning based on depth data," in *Proc. of IEEE Int. Conf. on Robotics and Automation*, May 2013, pp. 3230–3235.

[2] C. Choi, Y. Taguchi, O. Tuzel, M.-Y. Liu, and S. Ramalingam, "Voting-based pose estimation for robotic assembly using a 3d sensor," *2012 IEEE Int. Conf. on Robotics and Automation*, pp. 1724–1731, May 2012.

[3] C. Papazov, S. Haddadin, S. Parusel, K. Krieger, and D. Burschka, "Rigid 3d geometry matching for grasping of known objects in cluttered scenes," *The Int. Journal of Robotics Research*, vol. 31, no. 4, pp. 538–553, 2012.

[4] A. Agrawal, Y. Sun, J. Barnwell, and R. Raskar, "Vision guided robot system for picking objects by casting shadows," *Int. Journal of Robotics Research*, vol. 29, pp. 155–173, 2010.

[5] A. Schyja, A. Hypki, and B. Kuhlenkötter, "A modular and extensible framework for real and virtual bin-picking environments," in *Proc. of Int. Conf. on Robotics and Automation*, May 2012, pp. 5246–5251.

[6] D. Fischinger and M. Vincze, "Learning grasps for unknown objects in cluttered scenes," in *Proc. of IEEE Int. Conf. on Robotics and Automation*, 2013, pp. 601–608.

[7] M. Richtsfeld and M. Zillich, "Grasping unknown objects based on 2 1/2 d range data," in *Proc. of IEEE Int. Conf. on Automation Science and Engineering*, August 2008, pp. 691–696.

[8] S. Gottschalk, M. C. Lin, and D. Manocha, "Robust and accurate polygon interference detection," http://gamma.cs.unc.edu/OBB/, Version 2.00 [released June 10, 1997].

[9] D. Fiedler and H. Müller, "Impact of thermal and environmental conditions on the kinect sensor," in *Proc. of Int. Wksp on Depth Image Analysis at the 21st Int. Conf. on Pattern Recognition*, 2012.

[10] J.-D. Dessimoz, J. R. Birk, R. B. Kelley, H. A. S. Martins, and C. Lin, "Matched filters for bin picking," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 686–97, June 1984.

[11] O. Fischler and R. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Comm. of the Assoc. for Computing Machinery*, pp. 381–395, 1981.

[12] D. Kubus, T. Kröger, and F. M. Wahl, "On-line rigid object recognition and pose estimation based on inertial parameters," in *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2007, pp. 1402–1408.

[13] ——, "On-line estimation of inertial parameters using a recursive total least-squares approach," in *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2008, pp. 3845–3852.

[14] C. An, C. G. Atkeson, and J. M. Hollerbach, *Model-based Control of a Robot Manipulator*. The MIT Press, 1988.

[15] K. Kozlowski, *Modelling and Identification in Robotics*. Springer, 1998.

[16] L. Ljung, *System Identification*. Prentice Hall, 1999.

[17] J. J. Craig, *Introduction to Robotics*, 3rd ed. Prentice Hall, 2003.

[18] S. k. Zhou and R. Chellappa, "From sample similarity to ensemble similarity: Probabilistic distance measures in reproducing kernel hilbert space," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 28, no. 6, pp. 917–929, 2006.

[19] S. Julier, "The scaled unscented transformation," in *Proc. of American Control Conf.*, vol. 6, 2002, pp. 4555–4559.

[20] T. Kröger, *On-Line Trajectory Generation in Robotic Systems*, 1st ed. Berlin, Heidelberg, Germany: Springer, January 2010, vol. 58.

[21] I. Weidauer, D. Kubus, and F. M. Wahl, "A hierarchical extension of manipulation primitives and its integration into a robot control architecture," in *Accepted at IEEE International Conference on Robotics and Automation*, 2014.

[22] D. Kubus, A. Sommerkorn, T. Kröger, J. Maass, and F. Wahl, "Low-level control of robot manipulators," in *Wksp on Innovative Robot Control Architectures, IEEE Int. Conf. on Robotics and Automation*, 2010, pp. 38–45.

[23] B. Finkemeyer, T. Kröger, D. Kubus, M. Olschewski, and F. M. Wahl, "MiRPA: Middleware for robotic and process control applications," in *Wksp on Measures and Procedures for the Evaluation of Robot Architectures and Middleware, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, San Diego, CA, USA, October 2007, pp. 78–93.