A Side of Data with My Robot: Three Datasets for Mobile Manipulation in Human Environments

Matei Ciocarlie, *Member, IEEE,* Caroline Pantofaru, *Member, IEEE,* Kaijen Hsiao, *Member, IEEE,*
Gary Bradski, *Member, IEEE,* Peter Brook, and Ethan Dreyfuss

***Abstract*—Consideration of dataset design, collection and distribution methodology is becoming increasingly important as robots move out of fully controlled settings, such as assembly lines, and into unstructured environments. Extensive knowledge bases and datasets will potentially offer a means of coping with the variability inherent in the real world.**

In this study, we introduce three new datasets related to mobile manipulation in human environments. The first set contains a large corpus of robot sensor data collected in typical office environments. Using a crowd-sourcing approach, this set has been annotated with ground-truth information outlining people in camera images. The second dataset consists of three-dimensional models for a number of graspable objects commonly encountered in households and offices. Using a simulator, we have identified on each of these objects a large number of grasp points for a parallel jaw gripper. This information has been used to attempt a large number of grasping tasks using a real robot. The third dataset contains extensive proprioceptive and ground truth information regarding the outcome of these tasks.

All three datasets presented in this paper share a common framework, both in software (the Robot Operating System) and in hardware (the Personal Robot 2). This allows us to compare and contrast them from multiple points of view, including data collection tools, annotation methodology, and applications.

I. Introduction

Unlike its counterpart from the factory floor, a robot operating in an unstructured environment can expect to be confronted by the unexpected. Generality is an important quality for robots intended to work in typical human settings. Such a robot must be able to navigate around and interact with people, objects and obstacles in the environment, with a level of generality reflecting typical situations of daily living or working. In such cases, an extensive knowledge base, containing and possibly synthesizing information from multiple relevant scenarios, can be a valuable resource for robots aiming to cope with the variability of the human world.

Recent years have seen a growing consensus that one of the keys to robotic applications in unstructured environments lies in collaboration and reusable functionality [1], [2]. A result has been the emergence of a number of platforms and frameworks for sharing operational "building blocks," usually in the form of code modules, with functionality ranging from low-level hardware drivers to complex algorithms. By using a set of now well-established guidelines, such as stable, documented interfaces and standardized communication protocols, this type of collaboration has accelerated development towards complex applications. However, a similar set of methods for sharing and reusing data has been slower to emerge.

M. Ciocarlie, C. Pantofaru, K. Hsiao and G. Bradski are with Willow Garage Inc., Menlo Park, CA.
P. Brook is with the University of Washington, Seattle, WA.
E. Dreyfuss is with Redwood Systems, Fremont, CA.

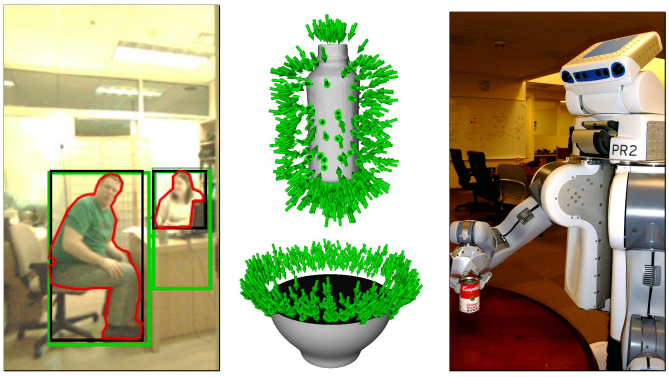


Fig. 1. Examples from the datasets presented in this study. **Left:** section of a camera image annotated with people's locations and outlines. **Middle:** 3D models of household objects with grasp point information (depicted by green arrows) generated in simulation. **Right:** the PR2 robot executing a grasp while recording visual and proprioceptive information.

In this paper, we present three datasets that utilize the same robot framework, comprised of the Robot Operating System (ROS) [1], [3] and the Personal Robot 2 (PR2) platform [4]. While sharing the underlying software and hardware architecture, they address different components of a mobile manipulation task: interacting with humans and grasping objects. They also highlight some of the different choices available for creating and using datasets for robots. As such, this comparison endeavors to begin a dialog on the format of datasets for robots. The three datasets, exemplified in Fig. 1, are:

- the **Moving People, Moving Platform** Dataset, containing robot perception data in office environments with an emphasis on person detection.
- the **Household Objects and Grasps** Dataset, containing 3D models of objects common in household and office environments, as well as a large set of grasp points for each model pre-computed in a simulated environment.
- the **Grasp Playpen** Dataset, containing both proprioceptive data from the robot's sensors and ground truth information from a human operator as the robot performed a large number of grasping tasks.

While new datasets can be made available independently of code or application releases, they can provide stable interfaces for algorithm development. The intention is not to tie datasets to specific code instances. Rather, both the dataset and the code can follow rigorous (yet possibly independent) release cycles, while explicitly tagging compatibility between specific versions (*e.g.*, Algorithm 1.0 has been trained on Data 3.2). The potential benefits of using such a release model for datasets include the following:

- defining a stable interface to the dataset component of

a release will allow external researchers to provide their own modified and/or extended versions of the data to the community, knowing that it will be directly usable by anyone running the algorithmic component;
- similarly, a common dataset and interface can enable direct comparison of multiple algorithms (*e.g.* [5]);
- a self-contained distribution, combining a compatible code release and the sensor data needed to test and use them, can increase research and development community by including groups who do not have access to hardware platforms.

The number of mobile manipulation platforms capable of combining perception and action is constantly rising; as a result, the methods by which we choose to share and distribute data are becoming increasingly important. In an ideal situation, a robot confronted with an unknown scenario will be able to draw on similar experiences from a different robot, and then finally contribute its own data back to the community. The **context** for this knowledge transfer can be on-line (with the robot itself polling and then sending data back to a repository), or off-line (with centralized information from multiple robots used as training data for more general algorithms). Other choices include the **format** and contents of the data itself (which can be raw sensor data or the result of task-specific processing), the source of **annotations** and other meta-data (expert or novice human users, or automated processing algorithms), *etc*. These choices will become highly relevant as we move towards a network of publicly accessible knowledge repositories for robots and their programmers. We will return to this discussion at the end of the study, after presenting each of the three datasets in more detail in the following sections.

II. The Moving People, Moving Platform Dataset

Personal robots operate in environments populated by people. They can interact with people on many levels, by planning to navigate towards a person, by navigating to avoid a specific person, by navigating around a crowd, by performing coordinated manipulation tasks such as object hand-off, or by avoiding contact with a person in a tabletop manipulation scenario. For all of these interactions to be successful, people must be perceived in an accurate and timely manner.

Training and evaluating perception strategies requires a large amount of data. This section presents the Moving People, Moving Platform Dataset [6], which contains robot sensor data of people in office environments. This dataset is available at http://bags.willowgarage.com/downloads/people_dataset.html.

The dataset is intended for use in offline training and testing of multi-sensor person detection and tracking algorithms that are part of larger planning, navigation and manipulation systems. Typical distances between the people and the robot are in the range of 0.5m to 5m. Thus, the data is more interesting for navigation scenarios such as locating people with whom to interact, than in table-top manipulation scenarios.

A. Related Work

The main motivation for creating this dataset was to encourage research into indoor, mobile robot perception of people. There is a large literature in the computer vision community on detecting people outdoors, from cars, in surveillance imagery, or in still images and movies on the Internet. Examples of such datasets are described below. In contrast, personal robots often function indoors. There is currently a lack of multi-modal data for creating and evaluating algorithms for detecting people indoors from a mobile platform. This is the vacuum the Moving People, Moving Platform Dataset aims to fill.

Two of the most widely used datasets for detecting and segmenting people in single images from the Internet are the INRIA Person Dataset [7] and the PASCAL Visual Object Challenge dataset [5]. Both datasets contain a large number of images, as well as bounding boxes annotating the extent of each person. The PASCAL dataset also contains precise outlines of each person. Neither dataset, however, contains video, stereo, or any other sensor information commonly available to robots. The people are contained in extremely varied environments (indoors, outdoors, in vehicles, etc.) People in the INRIA dataset are in upright poses referred to as "pedestrians" (e.g. standing, walking, leaning, etc.) On the other hand, poses in the PASCAL dataset are unrestricted. For the office scenarios considered in this paper, people are often not pedestrians. However, their poses are also not random.

Datasets of surveillance data, including [8] and the TUM Kitchen Dataset [9], are characterized by stationary cameras, often mounted above people's heads. Algorithms using these datasets make strong use of background priors and subtraction.

Articulated limb tracking is beyond the scope of this paper but should be mentioned. Datasets such as the CMU MoCap dataset [10] and HumanEva-II dataset [11] are strongly constrained by a small environment, simple background, and in the case of the CMU dataset, tight, uncomfortable clothing.

Detecting people from cars has been a focus in the research community of late. The Daimler Pedestrian Dataset [12] and Caltech Pedestrian Dataset [13] contain monocular video data taken from cameras attached to car windshields. Pedestrians are annotated with bounding boxes denoting the visible portions of their bodies, as well as bounding boxes denoting the predicted entire extent of their bodies, including occluded portions. In contrast to our scenario, the people in this dataset are pedestrians outdoors, and the cameras are moving quickly in the cars. Similarly to our scenario, the sensor is mounted in a moving platform.

In contrast to the above examples, the Moving People, Moving Platform dataset contains a large amount of data of people in office environments, indoors, in a realistic variety of poses, wearing their own clothing, taken from multiple sensors onboard a moving robot platform.

B. Contents and Collection Methodology

1) Collection Methodology: Datasets can be collected in many ways, and the collection methodology has an impact on both the type of data available and its accuracy. For the Moving People, Moving Platform Dataset, data was collected by tele-operating the PR2 to drive through 4 different office environments, recording data from onboard sensors. The robot's physical presence in the environment affected the data collected.

Tele-operation generates a different dataset than autonomous robot navigation; however, it was a compromise required to obtain entry into other companies' offices. Tele-operation also allowed online decisions about when to start and pause data collection, limiting dataset size and avoiding repetitive data such as empty hallways. However, it also opened the door to operator bias.

During collection, the subjects were asked to go about their normal daily routine. The approaching robot could be clearly heard, and so could not take people by surprise. Some people ignored the robot, while others were distracted by the novelty and stopped to take photographs or talk to the operator. The operator minimized tainting of the data, although some images of people with camera-phones were included for realism (as this scenario often occurs at robot demos).

Capturing natural human behavior is difficult, as discussed in [14]. A novel robot causes unnatural behavior (such as photo-taking) but is entertaining, and people are patient. On the other hand, as displayed toward the end of our data collection sessions, a robot cohabitating with humans for an extended time allows more natural behavior to emerge, but the constant monitoring presence leads to impatience and annoyance.

2) Contents - Robot Sensor Data: Given that this dataset is intended for offline training and testing, dataset size and random access speed are of minimal concern. In fact, providing as much raw data as possible is beneficial to algorithm development. The raw sensor data was therefore stored in ROS-format "bag" files [3]. The images contain Bayer patterns and are not rectified, the laser scans are not filtered for shadow points or other errors, and the image de-Bayering and rectification information is stored with the data. ROS bags make it easy to visualize, process and run data in simulated real-time within a ROS system. The following list summarizes the dataset contents, with an example in Figure 3. Figure 2 shows the robot's sensors used for dataset collection.

- A total of 2.5 hours of data in 4 different indoor office environments.
- 70 GB of compressed data (118 GB uncompressed)
- Images from wide field-of-view (FOV), color stereo cameras located approximately 1.4m off the ground, at 25Hz (640x480).
- Images from narrow FOV, monochrome stereo cameras located approximately 1.4m off the ground, at 25Hz (640x480).
- Bayer pattern, rectification and stereo calibration information for each stereo camera pair.
- Laser scans from a planar laser approximately 0.5ft off the ground, with a frequency of 40Hz.
- Laser scans from a planar laser on a tilting platform approximately 1.2m off the ground, at 20Hz.
- The robot's odometry and transformations between robot coordinate frames.

While raw data in a robotics-specific format like ROS bags is preferred by the robotics community, it is valuable to consider other research communities who may contribute solutions. For example, the computer vision community pursues research into person detection that is applicable to robotics

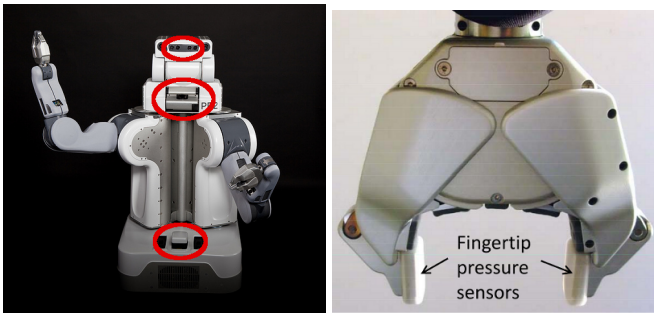
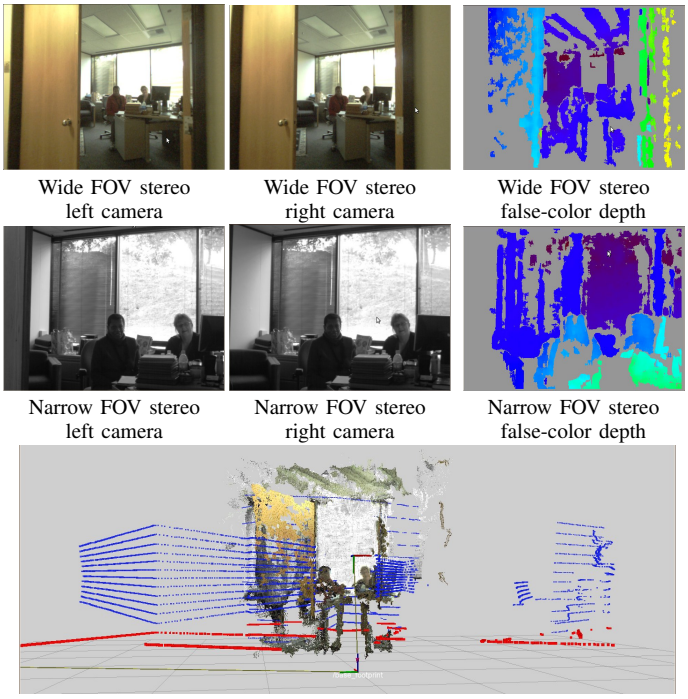


Fig. 2. **Left:** PR2 robot with sensors used for collecting the Moving People, Moving Platform dataset circled in red. From top to bottom: the wide FOV stereo camera pair and the narrow FOV stereo camera pair interleaved on the "head", the tilting 2D laser, and the planar 2D laser atop the robot's base. **Right:** the PR2 gripper and the tactile sensors used for collecting data during grasp execution.



3D visualization
Red/green/blue axes: robot's base and camera frames.
Red dots: data from the planar laser on the robot base.
Blue dots: 0.5 seconds of scans from the tilting laser.
The true-color point clouds are from the stereo cameras.

Fig. 3. A snapshot of data in the Moving People, Moving Platform dataset.

scenarios. To encourage participation in solving this robotics challenge, the dataset is also presented in a format familiar to the vision community: PNG-format images. In the current offering of the dataset, the PNG images are de-Bayered and rectified to correspond to the annotations (which will be discussed in the next section); however, they could also be offered in their raw form.

C. Annotations and Annotation Methodology

1) Annotation: All annotations in the dataset correspond to a de-Bayered, rectified version of the image from the left camera of the wide FOV stereo camera pair. Approximately one third of the frames were annotated, providing

Fig. 4. Examples of ground truth labels in the Moving People, Moving Platform Dataset. The images have been manipulated to improve outline visibility; they are brighter and have less contrast than the originals. The green bounding box is the predicted full extent of the person. The black bounding box corresponds to the visible portion of the person. The red polygon is an accurate outline of the visible portion of the person.

	Number of Images		
	Total	**Labeled**	**W/People**
Training files	57754	21064	13417
Testing files	50370	16646	-
Total	108124	37710	-

TABLE I
CONTENTS OF THE MOVING PEOPLE, MOVING PLATFORM DATASET.

Fig. 5. The Mechanical Turk interface for annotating outlines of people for the Moving People, Moving Platform Dataset. Workers were presented with the original image with a bounding box annotation of one person (by another worker) on the left, and an enlarged view of the bounding box on the right. The worker drew a polygonal outline of the person in the right-hand image.

approximately 38,000 annotated images. Table I presents the annotation statistics. Annotations take one of three forms: exact outlines of the visible parts of people, bounding boxes of the visible parts computed from the outlines, and bounding boxes of the predicted full extent of people, including occluded parts. Annotation examples can be found in Figure 4. These design decisions were driven by the desire for consistency with previous computer vision datasets, as well as the restrictions imposed by the use of Amazon's Mechanical Turk marketplace for labeling, which will be discussed in the next subsection.

Within the dataset ROS bags, annotations are provided as ROS messages, time-synchronized with their corresponding images. In order to align an annotation with an image, the user must de-Bayer and rectify the images. Since the annotations were created on the rectified images, the camera parameters may not be changed after annotation, but the algorithm used for de-Bayering may be improved. In addition, to complement the non-ROS dataset distribution, XML-format annotations are provided with the single image files.

2) Annotation Methodology: Annotation of the dataset was crowd-sourced using Amazon's Mechanical Turk marketplace [15]. The use of an Internet workforce allowed a large dataset to be created relatively quickly, but also had implications for the annotations. The workers were untrained and anonymous. Untrained workers are most familiar with rectified, de-Bayered images, and so the robot sensor data was presented as such. As discussed in the previous subsection, image-based annotations are generally incomplete for a robotics application.

Two separate tasks were presented to workers. In the first task, workers were presented with a single image and asked, for each person in the image, to draw a box around the entire person, even if parts of the person were occluded in the image. The visible parts of the person were reliably contained within the outline; however, variability occurred in the portion of the bounding box surrounding occluded parts of the person. This variability could be seen between consecutive frames in the video. In the vast majority of cases, however, workers agreed on the general direction and location of missing body parts. For example, if a person in an image sat at a desk with their legs occluded by the desk, all of the annotations predicted that there were legs behind the desk, below the visible upper body, but the annotations differed in the position of the feet at the bottom of the bounding box.

In the second task, workers were required to draw an accurate, polygonal outline of the visible parts of a single person in an enlarged image. The workers were presented with both the original image and an enlarged image of the predicted bounding box of a single person (as annotated by workers in the first task). An example of the interface can be seen in Figure 5. As this task was more constrained, the resulting annotations had less variability.

Mechanical Turk is a large community of workers of varying skill and intent; hence, quality control of results is an important issue. Mechanical Turk allows an employer to refuse to pay or ban under-performing workers. These acts, however, are frowned upon by the worker community who communicates regularly through message boards, resulting in a decreased and angry workforce. Thus, it is important to avoid refusing payment or banning workers whenever possible. The following are lessons learned in our quest for accurate annotations.

Lesson 1: Interface design can directly improve annotation accuracy. For the outline annotations described in this paper, the workers were presented an enlarged view of the

person's bounding box. Small errors at this enlarged scale were irrelevant at the original image size.

Lesson 2: Clear, succinct instructions improve annotation quality. Workers often skim instructions, so pictures with examples of good and bad results are more effective than text.

Lesson 3: Qualification tests are valuable. Requiring workers to take a multiple choice test to qualify to work on a task improved annotation quality significantly. The simple tests for these tasks verified full comprehension of the instructions, and were effective tools for removing unmotivated workers.

Lesson 4: The effective worker pool for a task is small. For each of the two labeling tasks, each image annotation could be performed by a different worker, implying that hundreds of workers would complete the thousands of jobs. This hypothesis was incorrect: approximately twenty workers completed more than 95% of the work. It appears that workers mitigate training time by performing many similar jobs. This also implies that a workforce can be loyal, so it is worthwhile to train and treat them well, which leads to the final lesson.

Lesson 5: Personalized worker evaluation increases annotation quality. Initially, workers graded their peers' annotations. Unfortunately, since grading was an easier task than annotating, it attracted less motivated workers. In addition, loyal annotators were upset by the lack of personal feedback. Grading the graders does not scale, and failing to notice a malicious grader leads to numerous misgraded annotations. These facts encouraged us to grade the annotations personally and write lengthy comments to workers making consistent mistakes. The workers were extremely receptive to this approach, quickly correcting their mistakes, thus significantly reducing duplication of work. Overall, personalized feedback for the small number of workers reduced our own workload.

There are other ways to identify incorrect annotations; however, they were not applicable in this situation. For example, the reCAPTCHA-style [16] of presenting two annotations and grading the second based on the first assumes that the errors are consistent. For the annotation task in the Moving People, Moving Platform Dataset, however, errors resulted from misunderstanding the instructions for a particular image scenario (e.g., a person truncated by the image border). Unless both of the images presented contain the same scenario(s), the redundancy of having two images cannot be exploited.

D. Applications

This dataset is intended exclusively for offline training and testing of person detection and tracking algorithms from a robot perspective. The use of multiple sensor modalities, odometry and situational information is encouraged. Some possible components that could be tested using this dataset are face detection, person detection, human pose fitting, and human tracking. Examples of information beyond that offered by other datasets that could be extracted and used for algorithm training include the appearances of people in multiple robot sensors, typical human poses in office environments (e.g. sitting, standing), illumination conditions (e.g. heavily backlit offices with windows), scene features (e.g. ceilings, desks, walls), and how people move around the robot. This is just a small sample of the applications for this dataset.

E. Future Work

It is important to take a moment to discuss the possible constraints on algorithm design imposed by the annotation format and methodology. Two-dimensional outlines can only be accurate in the image orientation and resolution. Robots, however, operate in three dimensions. Given that stereo camera information is noisy, it is unclear how to effectively project information from a two-dimensional image into the three-dimensional world. The introduction of more reliable instantaneous-depth sensors may ameliorate this problem. However, even a device such as the Microsoft Kinect sensor [17] is restricted to one viewpoint. Algorithms developed on such a dataset can only provide incomplete information. A format for three-dimensional annotations that can be obtained from an untrained workforce is an open area of research.

Short-term work for this dataset will be focused on obtaining additional types of annotations. It would be informative to have semantic labels for the dataset such as whether the person is truncated, occluded, etc. and pose information such as whether the person is standing, sitting, etc. Future datasets may focus on perceiving people during interaction scenarios such as object hand-off. Additional data from new sensors, such as the Microsoft Kinect, would also enhance the dataset.

Finally, an additional interesting dataset could be constructed containing relationships between people and objects, including spatial relationships and human grasps and manipulations of different objects. Object affordances could enhance the other datasets described in this paper.

III. The Household Objects and Grasps Dataset

A personal robot's ability to navigate around and interact with people can be complemented by its ability to grasp and manipulate objects from the environment, aiming to enable complete applications in domestic settings. In this section we describe a dataset that is part of a complete architecture for performing pick-and-place tasks in unstructured (or semi-structured) human environments. The algorithmic components of this architecture, developed using the ROS framework, provide abilities such as object segmentation and recognition, motion planning with collision avoidance, and grasp execution using tactile feedback. For more details, we refer the reader to our paper describing the individual code modules as well as their integration [18]. The knowledge base, which is the main focus of this paper, contains relevant information for object recognition and grasping for a large set of common household objects.

The objects and grasps dataset is available in the form of a relational database, using the SQL standard. This provides optimized relational queries, both for using the data on-line and managing it off-line, as well as low-level serialization functionality for most major languages. Unlike the dataset described in the previous section, the Household Objects and Grasps set is intended for both off-line use during training stages and on-line use at execution time; in fact, our current algorithms primarily use the second of these options.

An alternative for using this dataset, albeit indirectly, is in the form of remote ROS services. A ROS application

typically consists of a collection of individual nodes, communicating and exchanging information. The TCP/IP transport layer removes physical restrictions, allowing a robot to communicate with a ROS node situated in a remote physical location. All the data described in this section is used as the back-end for publicly available ROS services running on a dedicated accessible server, using an API defined in terms of high-level application requirements (*e.g* grasp planning). Complete information for using this option, as well as regular downloads for local use of the same data, are available at http://www.ros.org/wiki/household_objects_database.

A. Related Work

The database component of our architecture was directly inspired by the Columbia Grasp Database (CGDB) [19], [20], released together with processing software integrated with the *GraspIt!* simulator [21]. The CGDB contains object shape and grasp information for a very large ($n = 7,256$) set of general shapes from the Princeton Shape Benchmark [22]. The dataset presented here is smaller in scope ($n = 180$), referring only to actual graspable objects from the real world, and is integrated with a complete manipulation pipeline on the PR2 robot.

While the number of grasp-related datasets that have been released to the community is relatively small, previous research provides a rich set of data-driven algorithms for grasping and manipulation. The problems that are targeted range from grasp point identification [23] to dexterous grasp planning [24], [25] and grasping animations [26], [27], to name only a few. In this study, we are primarily concerned with the creation and distribution of the dataset itself, and the possible directions for future similar datasets used as online or offline resources for multiple robots.

B. Contents and Collection Methodology

One of the guiding principles for building this database was to enable other researchers to replicate our physical experiments, and to build on our results. The database was constructed using physical objects that are generally available from major retailers (while this current release is biased towards U.S.-based retailers, we hope that a future release can include international ones as well). The objects were divided into three categories: for the first two categories, all objects were obtained from a single retailer (IKEA and Target, respectively), while the third category contained a set of household objects commonly available in most retail stores. Most objects were chosen to be naturally graspable using a single hand (*e.g.* glasses, bowls, and cans); a few were chosen as use cases for two-hand manipulation problems (*e.g.* power drills).

For each object, we acquired a 3D model of its surface (as a triangular mesh). To the best of our knowledge, no off-the-shelf tool exists that can be used to acquire such models for a large set of objects in a cost- and time-effective way. To perform the task, we used two different methods, each with its own advantages and limitations:

- for those objects that are rotationally symmetric about an axis, we segmented a silhouette of the object against a known background, and used rotational symmetry to generate a complete mesh. This method can generate high-resolution, very precise models, but is only applicable to rotationally symmetrical objects.
- for all other objects, we used the commercially available tool 3DSOM (Creative Dimension Software Ltd., U.K.). 3DSOM builds a model from multiple object silhouettes, and can not resolve object concavities and indentations.

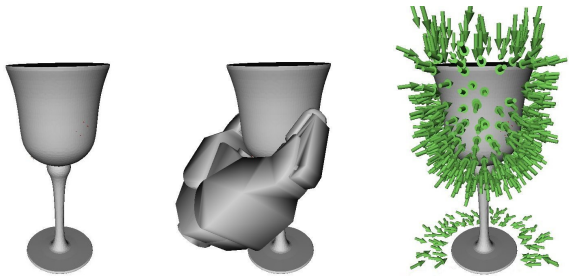


Fig. 6. Grasp planning in simulation on a database model. Left: the object model; Middle: grasp example using the PR2 gripper; Right: the complete set of pre-computed grasps for the PR2 gripper. Each arrow shows one grasp: the arrow location shows the position of the center of the leading face of the palm, while its orientation shows the gripper approach direction. Gripper "roll" around the approach direction is not shown.

Overall, for each object, the database contains the following core information:

- the maker and model name (where available);
- the product barcode (where available);
- a category tag (*e.g.* glass, bowl, *etc.*);
- a 3D model of the object surface, as a triangular mesh.

For each object in the database, we used the *GraspIt!* simulator to compute a large number of grasp points for the PR2 gripper (shown in Figure 2). We note that, in our current release, the definition of a good grasp is specific to this gripper, requiring both finger pads to be aligned with the surface of the object (finger pad surfaces contacting with parallel normal vectors) and further rewarding postures where the palm of the gripper is close to the object as well. In the next section, we will discuss a data-driven method for relating the value of this quality metric to real-world probability of success for a given grasp.

Our grasp planning tool used a simulated annealing optimization, performed in simulation, to search for gripper poses relative to the object that satisfied this quality metric. For each object, this optimization was allowed to run over 4 hours, and all the grasps satisfying our requirements were saved in the database; an example of this process is shown in Figure 6 (note that the stochastic nature of our planning method explains the lack of symmetry in the set of database grasps, even in the case of a symmetrical object). This process resulted in an average of 600 grasp points for each object. In the database, each grasp contains the following information:

- the pose of the gripper relative to the object;
- the value of the gripper degree of freedom, determining the gripper opening;
- the value of the quality metric used to distinguish good grasps.

The overall dataset size, combining both model and grasp

Fig. 7. The PR2 robot performing a grasping task on an object recognized from the model database.

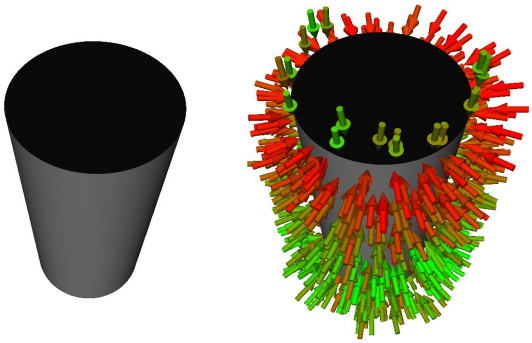


Fig. 8. (Best seen in color) Quantifying grasp robustness to execution errors, from low (red markers) to high (green markers). Note that grasps in the narrow region of the cup are seen as more robust to errors, as the object fits more easily within the gripper.

information, is 76MB uncompressed and 12MB compressed.

C. Annotations and Annotation Methodology

Unlike the other two datasets presented in this paper, the models and grasps set does not contain any human-generated information. However, grasp points derived using our autonomous algorithm have one important limitation: they do not take into account object-specific semantic information or intended use. This could mean a grasp that places one finger inside a cup or a bowl, or prevents a tool from being used. In order to alleviate this problem, an automated algorithm could take into account more recent methods for considering intended object use [28]. Alternatively, a human operator could be used to demonstrate usable grasps [29]. The scale of the dataset, however, precludes the use of few expert operators, while a crowd-sourcing approach, similar to one discussed in the previous section in the context of labeling persons, raises the difficulty of specifying 6-dimensional grasp points with simple input methods such as a point-and-click interface.

D. Applications

The database described in this study was integrated in a complete architecture for performing pick and place tasks on the PR2 robot. A full description of all the components used for this task is beyond the scope of this paper. We present here a high-level overview with a focus on the interaction with the database; for more details on the other components, we refer the reader to [18].

In general, a pick-and-place task begins with a sensor image of the object(s) to be grasped, in the form of a point cloud acquired using a pair of stereo cameras. Once an object is segmented, a recognition module attempts to find a match in the database, using an iterative matching technique similar to the ICP algorithm [30]. We note that this recognition method only uses the 3D surface models of the objects stored in the database. Our data-driven analysis discussed in the next section has also been used to quantify the results of this method and relate the recognition quality metric to ground truth results.

If a match is found between the target object and a database model, a grasp planning component will query the database for all pre-computed grasp points of the recognized object. Since these grasp points were pre-computed in the absence of other obstacles and with no arm kinematic constraints, an additional module checks each grasp for feasibility in the current environment. Once a grasp is deemed feasible, the motion planner generates an arm trajectory for achieving the grasp position, and the grasp is executed. An example of a grasp executed using the PR2 robot is shown in Figure 7. For additional quantitative analysis of the performance of this manipulation framework, we refer the reader to [18].

The manipulation pipeline can also operate on novel objects. In this case, the database-backed grasp planner is replaced by an on-line planner able to compute grasp points based only on the perceived point cloud from an object; grasps from this grasp planner are used in addition to the pre-computed grasps to generate the Grasp Playpen database described in the next section. Grasp execution for unknown objects is performed using tactile feedback in order to compensate for unexpected contacts. We believe that a robot operating in an unstructured environment should be able to handle unknown scenarios while still exploiting high-level perception results and prior knowledge when these are available. This dual ability also opens up a number of promising avenues for autonomous exploration and model acquisition that we will discuss below.

E. Future Work

We believe that the dataset that we have introduced, while useful for achieving a baseline for reliable pick and place tasks, can also serve as a foundation for more complex applications. Efforts are currently underway to:

- improve the quality of the dataset itself, *e.g.* by using 3D model capture methods that can correctly model concavities or model small and sharp object features at better resolution;
- improve the data collection process, aiming to make it faster, less operator-intensive, or both;
- use the large computational budgets afforded by off-line execution to extract more relevant features from the data, which can in turn be stored in the database;
- extend the dataset to include grasp information for some of the robotic hands most commonly used in the research community;
- develop novel algorithms that can make use of this data at runtime;

- improve the accessibility and usability of the dataset for the community at large.

One option for automatic acquisition of high-quality 3D models for a wide range of objects is to use high-resolution stereo data, able to resolve concavities and indentations, in combination with a pan-tilt unit. Object appearance data can be extended to also contain 2D images, from a wide range of viewpoints. This information can then be used to pre-compute relevant features, both two- and three-dimensional, such as SURF [31], PFH [32] or VFH [33]. This will enable the use of more powerful and general object recognition methods.

The grasp planning process outlined here for the PR2 gripper can be extended to other robot hands as well. For more dexterous models, a different grasp quality metric can be used, taking into account multi-fingered grasps, such as metrics based on the Grasp Wrench Space [34]. The Columbia Grasp Database also shows how large scale off-line grasp planning is feasible even for highly dexterous hands, with many degrees of freedom [19].

The grasp information contained in the database can be exploited to increase the reliability of object pickup tasks. An example of relevant off-line analysis is the study of how each grasp in the set is affected by potential execution errors, stemming from imperfect robot calibration or incorrect object recognition or pose detection. Our preliminary results show that we can indeed rank grasps by their robustness to execution errors; an example is shown in Figure 8. In its current implementation, this analysis is computationally intensive, but it can be performed off-line and the results stored in the database for online use.

IV. The Grasp Playpen Dataset

Using the pick and place architecture described in the previous section, we have set up a framework that we call the "Grasp Playpen" for evaluating grasps of objects using the PR2 gripper, and recording relevant data throughout the entire process. In this framework, the robot performed grasps of objects from the Household Objects dataset placed at known locations in the environment, enabling us to collect ground truth information for object shape, object pose, and grasp attempted. Furthermore, the robot attempted to not only grasp the object, but also shake it and transport it around in an attempt to estimate how robust the grasp is. Such data is useful for offline training, testing, and parameter estimation for both object recognition and grasp planning and evaluation algorithms. The Grasp Playpen dataset can be downloaded for use at http://bags.willowgarage.com/downloads/grasp_playpen_dataset/grasp_playpen_dataset.html.

A. Related Work

Although there has been a significant amount of research that uses data from a large number of grasps to either learn how to grasp or evaluate grasp features, it has generally not been accompanied by releases of the data itself. For instance, Balasubramanian *et al.* [35] use a similar procedure of grasping and shaking objects to evaluate the importance of various features used in grasp evaluation such as orthogonality. Detry *et al.* [36] execute a large number of grasps with a robot in order to refine estimated grasp affordances for a small number of objects. However, none of the resulting data appears to be publicly available. Saxena *et al.* [23] have released a labeled training set of images of objects labeled with the 2-d location of the grasping point in each image; however, the applicability of such data is limited. The Semantic Database of 3D Objects from TU Muenchen [37] contains point cloud and stereo camera images from different views for a variety of objects placed on a rotating table, but the objects are not meshed and the dataset contains no data related to grasping.

B. Contents and Collection Methodology

Each grasp recording documents one attempt to pick up a single object in a known location, placed alone on a table, as on the right side of Figure 1. The robot selects a random grasp by either: 1) trying to recognize the object on the table and using a grasp from the stored set of grasps for the best detected model in the Household Objects and Grasps database (planned using the *GraspIt!* simulator), or 2) using a grasp from a set generated by the novel-object grasp planner based on the point cloud. It then tries to execute the grasp. To estimate the robustness of the grasp chosen, the robot first attempts to lift the object off the table. If that succeeds, it will slowly rotate the object into a sideways position, then shake the object vigorously along two axes in turn, then move the object off away from the table and to the side of the robot, and finally attempt to place it back on the other side of the table. Visual and proprioceptive data from the robot is recorded during each phase of the grasp sequence; the robot automatically detects if and when the object is dropped, and stops both the grasp sequence and the recording.

In total, the dataset contains recordings of 490 grasps of 30 known objects from the Household Objects and Grasps Dataset, collected using three different PR2 robots over a 3-week period. Most of these objects are shown in Figure 9. Each grasp recording includes both visual and proprioceptive data. The dataset also contains 150 additional images and point clouds of a total of 44 known objects from the Household Objects and Grasps Dataset, including the 30 objects used for grasping. An example point cloud with its ground-truth model mesh overlaid is shown in Figure 9. Recorded data is stored as ROS-format "bag" files, as in the Moving People, Moving Platform dataset. Each grasp also has an associated text file summarizing the phase of the grasp reached without dropping the object, as well as any annotations added by the person.

Each grasp recording contains visual data of the object on the table prior to the grasp, from two different views obtained by moving the head:

- Images and point clouds from the narrow FOV, monochrome stereo cameras (640x480)
- Images from the wide FOV, color stereo cameras (640x480)
- Images from the gigabit color camera (2448x2050)
- The robot's head angles and camera frames

During the grasp sequence, the recorded data contains:

- Narrow and wide FOV stereo camera images (640x480, 1 Hz)
- Grasping arm forearm camera images (640x480, 5 Hz)
- Grasping arm fingertip pressure array data (25 Hz)
- Grasping arm accelerometer data (33.3 kHz)
- The robot's joint angles (1.3 kHz)
- The robot's camera and link frames (100 Hz)
- The requested pose of the gripper for the grasp

The average size of all the recorded data for one grasp sequence (compressed or uncompressed) is approximately 500 MB; images and point clouds alone are approximately 80 MB.

C. Annotations and Annotation Methodology

The most important annotations for this dataset contain the ground-truth model ID and pose for each object. Each object is placed in a randomly-generated, known location on the table by carefully aligning the point cloud for the object (as seen through the robot's stereo cameras) with a visualization of the object mesh in the desired location. The location of the object is thus known to within operator precision of placing the object, and is recorded as ground truth.

Further annotations to the grasps are added to indicate whether the object hit the table while being moved to the side or being placed, whether the object rotated significantly in the grasp or was placed in an incorrect orientation, and whether the grasp was stopped due to a robot or software error.

D. Applications

The recorded data from the Grasp Playpen Dataset is useful for evaluating and modeling the performance of object detection, grasp planning, and grasp evaluation algorithms.

For the ICP-like object detection algorithm described in section III-D, we have used the recorded object point clouds along with their ground-truth model IDs (and the results of running object detection) to create a model for how often we get a correct detection (identify the correct object model ID) for different returned values of the detection algorithm's "match error," which is the average distance between each stereo point cloud point and the proposed object's mesh. The resulting Naive Bayes model is shown in Figure 10, along with a smoothed histogram of the actual proportion of correct detections seen in the Grasp Playpen Dataset.

For the *GraspIt!* quality metric described in section III-B, we have used the grasps that were actually executed, along with whether they were successful or not (and *GraspIt!*'s estimated grasp quality for those grasps, based on the ground-truth model and pose), to model how often grasps succeed or fail in real life for different quality values returned by *GraspIt!*. Histogrammed data from the Grasp Playpen Dataset is shown in Figure 11, along with the piecewise-linear model for grasp quality chosen to represent it.

We have also used just the recorded object point clouds to estimate how well other grasp planners and grasp evaluation algorithms do on real (partial) sensor data. Because we have the ground truth model ID and pose, we can use a geometric simulator such as *GraspIt!* to estimate how good an arbitrary grasp is on the true object geometry. Thus, we can ask a new grasp planner to generate grasps for a given object point cloud, and then evaluate in *GraspIt!* how likely that grasp is to succeed (with energy values translated into probabilities via the model described above). Or we can generate grasps using any grasp planner or at random, and ask a new grasp evaluator to say how good it thinks each grasp is (based on just seeing the point cloud), and again use the ground truth model pose/geometry to compare those values to *GraspIt!*'s success probability estimates. This allows us to generate data on arbitrarily large numbers of grasps, rather than just the 490 recorded grasps; we have used this technique ourselves to evaluate new grasp planners and evaluators, as well as to create models for them and perform feature-weight optimization.

Fig. 9. (left) A subset of the objects used in the Grasp Playpen Dataset's grasp recordings. (right) The point cloud for a non-dairy creamer bottle, with the appropriate model mesh overlaid in the recorded ground-truth pose.

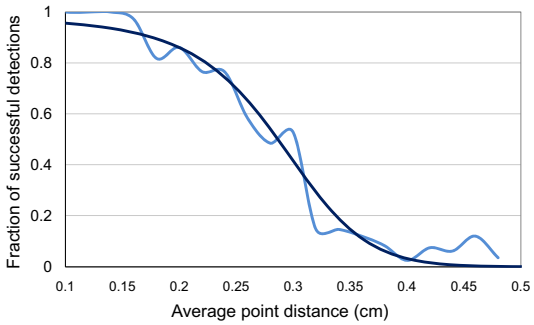


Fig. 10. Correct object recognition rates vs. the object detector's match error (average point distance) for our object recognizer. The blue line shows data from the grasp playpen dataset; the black line shows the Naive Bayes model chosen to approximate it.

E. Future Work

Because we use random grasps planned using our available grasp planners to grasp the objects presented to the robot, and because those grasps tend to be of high quality, approximately 90% of the grasps in the dataset succeed in at least lifting the object. Thus, although the data is useful for differentiating very robust grasps from only marginal grasps, we would require more data on grasp failures to better elucidate the difference between marginal and bad grasps. In the future, we plan to obtain data for more random/less good grasps. We also plan to obtain data for more complex/cluttered scenes than just single objects on a table.

Other planned or possible uses of the data include:

- testing object recognition and pose estimation algorithms;
- trying to predict when a collision has occurred based on the recorded accelerometer data from grasps in which the

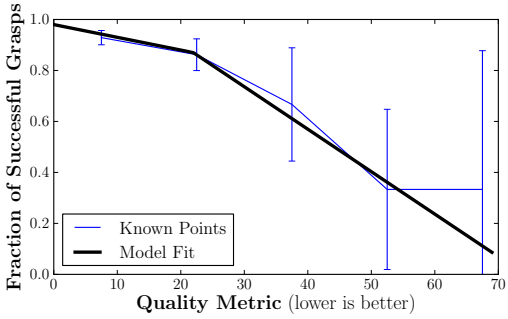


Fig. 11. Experimental grasp success percentages vs. *GraspIt!*'s grasp quality metric for the PR2 gripper. The blue line shows binned data from all 490 grasps in the grasp playpen dataset; the black line shows the piecewise-linear model chosen to approximate it. Blue error bars show 95% confidence on the mean, computed using bootstrap sampling.

object hit the table;
- testing in-hand object tracking algorithms;
- learning graspable features and weights for grasp features from image and point cloud data;

Obtaining grasp recordings by manually placing objects in the manner used for the Grasp Playpen Dataset is a fairly labor-intensive method. Killpack and Kemp have recently released code and the mechanical design for a PR2 playpen [38] that allows one to record grasps using the PR2 in a semi-automated fashion. Currently there is no mechanism for determining the ground-truth pose of the object being grasped, which is necessary for many of the proposed applications of the Grasp Playpen Dataset. However, automatically-generated grasp recordings, if done with objects with known models, could be annotated using Mechanical Turk, using a tool that allows a person to match and pose the correct object model.

V. Discussion and Conclusions

The datasets discussed in this paper are united by the ROS framework, their collection via the PR2 platform and their applicability to indoor home and office scenarios. The datasets' applications, however, force them to differ in multiple ways.

The Moving People, Moving Platform dataset is intended to be used in an off-line knowledge transfer **context**. In other words, robots are meant to utilize the data in batch format to train person-detection algorithms, and then once again in batch format to evaluate these algorithms. This off-line mechanism implies that access speed and dataset size are not of primary importance when considering the dataset **format** and contents. This allows the data to be presented in its raw, loss-less format. Off-line training is best performed with large amounts of data and annotation, and the nature of the annotations in this case required human input. These factors led to using humans in a crowd-sourced environment as a source of **annotations**. All of these requirements were met within the ROS framework by using ROS bag files and providing the data on the Internet for batch download.

The Household Objects and Grasps Dataset is primarily used in an on-line knowledge transfer **context**. This implies that the **format** and contents need to support fast random access, both in retrieving the data from the Internet and accessing individual data elements within the dataset. Thus, the data is stored in a relational database. The information is also compressed whenever possible, to grasp points or object meshes instead of full object images or scans. Computing grasp points appropriate to a robot is performed automatically and off-line using the *GraspIt!* simulator. No additional **annotations** from human sources are provided. The relational database containing this dataset has an interface within the ROS framework, allowing a running robot system to access the data on-line.

The Grasp Playpen Dataset provides an additional venue for grasp information, but this time the knowledge transfer is intended to happen in an off-line **context**. As in the Moving People, Moving Platform Dataset, the data does not need to be accessed quickly, and the size of the dataset is less important. This allows for storage in raw **format** in ROS bags, and the contents are less restricted, including images, point clouds, and additional sensor data for later exploration. Finally, given the broader potential uses of this dataset, the source of **annotations** is both automatic, generated by the robot as it successfully or unsuccessfully manipulates an object, and manual, with human annotations in text files. Once again, the data is available for batch download and can be viewed within the ROS framework.

The knowledge transfer context, the format and contents of the data, and the source of annotations are only some of the important characteristics of robotic datasets. We have expanded on them in this study as they are particularly relevant to the releases presented here; there are, however, a number of additional issues to consider when designing datasets. An incomplete list includes the following: are there other communities who could offer interesting input into the data, such as the computer vision community for the Moving People, Moving Platform dataset? What is the correct accuracy level? Can the dataset be easily expanded? Is it possible to add in additional sensor modalities or annotation modalities, perhaps in the way that the Grasp Playpen dataset extends the Household Objects and Grasps dataset? Does the data reflect the realistic conditions in which a scenario will be encountered? Can the objects in the Household Objects dataset be recognized in clutter, or do people normally act as they do in the Moving People dataset? Finally, does there need to be a temporal component to the data, such as people or objects appearing differently at night versus during the day? This is only a small sample of the questions which should be asked.

Dataset collection and annotation for mobile robots is typically a time and resource-intensive task, and the datasets presented here are no exception. Furthermore, obtaining such datasets requires access to a robot such as the PR2, which are not available to everyone. In light of the effort and resources required, we hope that by releasing these datasets, we can allow others to access useful data for their own research that they would not otherwise be able to obtain.

A particularly compelling direction of research considers the possibility of robots automatically augmenting and sharing datasets as they operate in their normal environments. People regularly draw on online information when faced with a new environment, getting data such as directions and product

information from ubiquitous mobile communication devices. In a similar way, robots can share their experiences in an online fashion, and some of the technology described in this paper can enable this exchange. For example, a robot can regularly collect sensor data from its surroundings, use a crowd-sourcing method to annotate it, and contribute it back to the Moving People, Moving Platform dataset.

The grasping pipeline presented here can serve as a foundation for fully automatic model acquisition: a robot can grasp a previously unseen object, inspect it from multiple viewpoints, and acquire a complete model, using techniques such as the ones presented in [39]. A robot could also learn from past pick-up trials. Additional meta-data, such as object classes, labels, or outlines in sensor data can be obtained on-line using a crowd-sourcing similar to the one used for the Moving People, Moving Platform dataset. Visual and proprioceptive information from any attempted grasp can be added to the Grasp Playpen set. Numerous other possibilities exist as we move towards a set of online resources for robots.

Dataset design is a complex subject, but collecting and presenting data in an organized and cohesive manner is key to progress in robotics. The datasets presented in this paper are a small step toward useful mobile manipulation platforms operating in human environments. By continuing to collect and distribute data in open formats such as ROS, a diverse array of future algorithms and robots can learn from experience.

REFERENCES

[1] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, "ROS: an open-source Robot Operating System," in *Intl. Conf. on Robotics and Automation*, 2009.

[2] P. Fitzpatrick, G. Metta, and L. Natale, "Towards long-lived robot genes," *Robotics and Autonomous Systems*, vol. 56, no. 1, pp. 29–45, 2008.

[3] "ROS Wiki," http://www.ros.org.

[4] Willow Garage, "The PR2," http://www.willowgarage.com/pages/pr2/overview.

[5] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *Intl. Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, June 2010. [Online]. Available: http://pascallin.ecs.soton.ac.uk/challenges/VOC/

[6] C. Pantofaru, "The Moving People, Moving Platform Dataset," http://bags.willowgarage.com/downloads/people_dataset.html, 2010.

[7] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[8] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua, "Multi-Camera People Tracking with a Probabilistic Occupancy Map," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 267–282, February 2008.

[9] M. Tenorth, J. Bandouch, and M. Beetz, "The TUM Kitchen Data Set of Everday Manipulation Activities for Motion Tracking and Action Recognition," in *IEEE Int'l Workshop on Tracking Humans for the Evaluation of their Motion in Image Sequences (THEMIS)*, 2009.

[10] "CMU Graphics Lab Motion Capture Database," http://mocap.cs.cmu.edu/.

[11] L. Sigal, A. Balan, and M. Black, "HumanEva: Synchronized Video and Motion Capture Dataset and Baseline Algorithm for Evaluation of Articulated Human Motion," *International Journal of Computer Vision*, vol. 87, 2010.

[12] M. Enzweiler and D. M. Gavrila, "Monocular Pedestrian Detection: Survey and Experiments," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 31, no. 12, pp. 2179–2195, 2009. [Online]. Available: http://www.gavrila.net/Research/Pedestrian_Detection/Daimler_Pedestrian_Benchmark_D/Daimler_Pedestrian_Detection_B/daimler_pedestrian_detection_b.html

[13] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian Detection: A Benchmark," in *IEEE Intl. Conf. on Computer Vision and Pattern Recognition*, June 2009. [Online]. Available: http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/

[14] C. Pantofaru, "User Observation & Dataset Collection for Robot Training," in *ACM/IEEE Conference on Human-Robot Interaction (HRI)*, 2011.

[15] "Amazon Mechanical Turk," https://www.mturk.com.

[16] L. von Ahn, B. Murer, C. McMillen, D. Abraham, and M. Blum, "reCAPTCHA: Human-Based Character Recognition via Web Security Measures," *Science*, vol. 321, pp. 1465–1468, September 2008.

[17] Microsoft Corp., "Kinect for Xbox 360."

[18] M. Ciocarlie, K. Hsiao, E. Jones, S. Chitta, R. B. Rusu, and I. A. Sucan, "Towards Reliable Grasping and Manipulation in Household Environments," in *Intl. Symp. on Experimental Robotics*, 2010.

[19] C. Goldfeder, M. Ciocarlie, H. Dang, and P. Allen, "The Columbia Grasp Database," in *Intl. Conf. on Robotics and Automation*, 2009.

[20] C. Goldfeder, M. Ciocarlie, J. Peretzman, H. Dang, and P. Allen, "Data-Driven Grasping with Partial Sensor Data," in *Intl. Conf. on Intelligent Robots and Systems*, 2009.

[21] A. Miller and P. K. Allen, "GraspIt!: A Versatile Simulator for Robotic Grasping," *IEEE Rob. and Autom. Mag.*, vol. 11, no. 4, 2004.

[22] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, "The Princeton Shape Benchmark," in *Shape Modeling and Applications*, 2004. [Online]. Available: http://dx.doi.org/10.1109/SMI.2004.1314504

[23] A. Saxena, J. Driemeyer, and A. Ng, "Robotic Grasping of Novel Objects using Vision," *International Journal of Robotics Research*, vol. 27, no. 2, pp. 157–173, 2008.

[24] A. Morales, T. Asfour, P. Azad, S. Knoop, and R. Dillmann, "Integrated Grasp Planning and Visual Object Localization for a Humanoid Robot with Five-Fingered Hands," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2006.

[25] Y. Li, J. L. Fu, and N. S. Pollard, "Data-Driven Grasp Synthesis Using Shape Matching and Task-Based Pruning," *IEEE Trans. on Visualization and Computer Graphics*, vol. 13, no. 4, pp. 732–747, 2007.

[26] Y. Aydin and M. Nakajima, "Database Guided Computer Animation of Human Grasping Using Forward and Inverse Kinematics," *Computers and Graphics*, vol. 23, 1999.

[27] K. Yamane, J. Kuffner, and J. Hodgins, "Synthesizing Animations of Human Manipulation Tasks," *ACM Transactions on Graphics*, vol. 23, no. 3, 2004.

[28] D. Song, K. Huebner, V. Kyrki, and D. Kragic, "Learning Task Constraints for Robot Grasping using Graphical Models," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, 2010.

[29] C. de Granville, J. Southerland, and A. Fagg, "Learning Grasp Affordances through Human Demonstration," in *Intl. Conf. on Development and Learning*, 2006.

[30] P. J. Besl and M. I. Warren, "A Method for Registration of 3-D Shapes," *IEEE Trans. on Pattern Analysis*, vol. 14, no. 2, pp. 239–256, 1992.

[31] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "SURF: Speeded Up Robust Features," *Computer Vision and Image Understanding*, vol. 110, no. 3, 2008.

[32] R. B. Rusu, N. Blodow, and M. Beetz, "Fast Point Feature Histograms (FPFH) for 3D Registration," in *Intl. Conf. on Robotics and Automation*, 2009. [Online]. Available: http://files.rbrusu.com/publications/Rusu09ICRA.pdf

[33] R. B. Rusu, G. Bradski, R. Thibaux, and J. Hsu, "Fast 3D Recognition and Pose Using the Viewpoint Feature Histogram," in *Intl. Conf. on Intelligent Robots and Systems*, 2010.

[34] C. Ferrari and J. Canny, "Planning Optimal Grasps," in *IEEE Intl. Conf. on Robotics and Automation*, 1992, pp. 2290–2295.

[35] R. Balasubramanian, L. Xu, P. Brook, J. Smith, and Y. Matsuoka, "Human-Guided Grasp Measures Improve Grasp Robustness on a Physical Robot," in *ICRA*, 2010.

[36] R. Detry, E. Baseski, M. Popovic, Y. Touati, N. Krueger, O. Kroemer, J. Peters, and J. Piater, "Learning Object-specific Grasp Affordance Densities," in *Intl. Conf. on Development and Learning*, 2009.

[37] M. Tenorth, J. Bandouch, and M. Beetz, "The TUM Kitchen Data Set of Everyday Manipulation Activities for Motion Tracking and Action Recognition," in *IEEE Int. Workshop on Tracking Humans for the Evaluation of their Motion in Image Sequences (THEMIS), held in conjunction with ICCV*, 2009.

[38] M. Killpack and C. Kemp, "ROS wiki page for the pr2_playpen package," http://www.ros.org/wiki/pr2_playpen.

[39] M. Krainin, P. Henry, X. Ren, and D. Fox, "Manipulator and Object Tracking for In Hand Model Acquisition," in *Intl. Conf. on Robotics and Automation*, 2010.