Chapter 1
The UBC Visual Robot Survey: A Benchmark for Robot Category Recognition

David Meger and James J. Little
The University of British Columbia
201-2366 Main Mall
Vancouver, Canada
V6T 1Z4
e-mail: {dpmeger,little}@cs.ubc.ca

Abstract Recognizing objects is a fundamental capability for robotic systems but comparing algorithms on similar testing situations remains a challenge. This makes characterizing the current state-of-the-art difficult and impedes progress on the task. We describe a recently proposed benchmark for robotic object recognition, named the UBC Visual Robot Survey, which is a robot-collected dataset of cluttered kitchen scenes. The dataset contains imagery and range data collected from a dense sampling of viewpoints. Objects have been placed in realistic configurations that result in clutter and occlusion, similar to common home settings. This data and accompanying tools for simulation from real data enable the study of robotic recognition methods. They specifically allow focus on specific concerns in robotics such as spatial evidence integration and active perception. We describe the method used to produce the dataset in detail, a suite of testing protocols and the current state-of-the-art performance on the dataset.

1.1 Robot Object Category Recognition

The locations and semantic labels of objects in the world are essential for robots during many real-world tasks. Estimating this information from a platform's sensors, known as *robot object category recognition*, is a challenging task due to: the wide variety of objects that share a semantic label; the indirect connections between object labels and raw sensory data (typically features are extracted and passed through a classification function); as well as clutter and occlusion in the world that leads to missing information. There are several uniquely robotic aspects to the problem, including the need to actively move the robot's sensors in order to obtain informative viewpoints and the possibility to fuse information across sensing modalities and spatial locations.

This paper describes a recently-established evaluation benchmark specifically tailored to the robot recognition problem, named the University of British Columbia

Visual Robot Survey (UBC VRS). It has been motivated by the example of object recognition in Computer Vision, where rapid progress has been made through standardization around the Pascal Visual Object Classes (VOC) challenge [7], a high participation, yearly contest of ever-increasing difficulty, and benchmark tasks for distinguishing large numbers of object categories, such as Caltech 101 [8] and 256 [11]. Several robotics challenges exist, including the Semantic Robot Vision Challenge (SRVC) [1] and Solutions in Perception Challenge [2], which compare near real-time systems on robot recognition tasks at a particular venue once per year. These contests capture the full scope of robot recognition, but the requirement to travel to the contest location in order to participate limits their accessibility.

Several datasets based on RGB-D data such as that available from the Microsoft Kinect have recently been released. For example, the Berkeley 3D Object Dataset [14] is composed of many indoor scenes contributed by the community through *crowd-sourcing* and annotated by humans. While there are more images and more object types in this dataset than the one we present, each scene is captured from only a single viewpoint, which does not allow exploration of recognition methods involving robot motion. The Multi-View RGB-D Object Dataset by Lai *et al.* [15] includes a large number of scenes containing a single object on a turn-table, captured with an image-depth sensor from a number of viewpoints, as well as a smaller number of scenes containing multiple objects captured with hand-held trajectories. This dataset allows for rapid iteration and direct comparison between methods, but the single trajectory through each scenes precludes active perception.

The contribution of the UBC VRS evaluation benchmark is to allow the unique aspects of the robot recognition problem to be explored with statistical significance and repeatability. These aspects include: the use of 3D and visual sensory data; the ability to actively control the robot's path and influence the series of images obtained; and the challenge of cluttered scenes present in real environments.

1.2 UBC Visual Robot Survey Benchmark

While performing active perception, a robot moves through its 3D environment, controlling its own position as well as the orientation of its sensors. We have attempted to capture all information necessary to simulate (with real data) this perceptual experience for recognition systems at both training and test time. To this end, our database is created by recording the sensory experience of a physical robot following a trajectory that passes through a dense sampling of poses within a number of environments. The poses are registered to a consistent coordinate frame using a visual fiducial target of known geometry. A human manually annotates the locations of all object instances from several categories, both in the 3D coordinate frame and within each collected image. Figure 1.1 illustrates the final product of this procedure, which is robot sensor data from a set of viewpoints of each scene, along with geometric knowledge linking that data into a common frame, and annotations both in 3D and 2D.

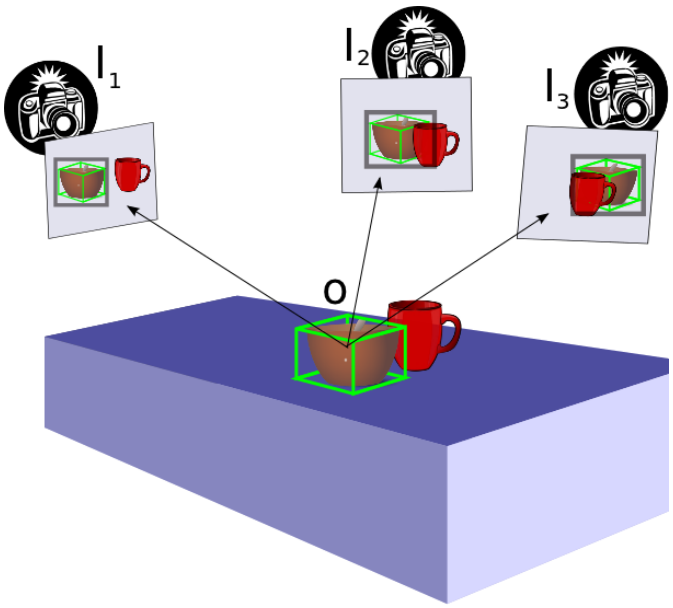
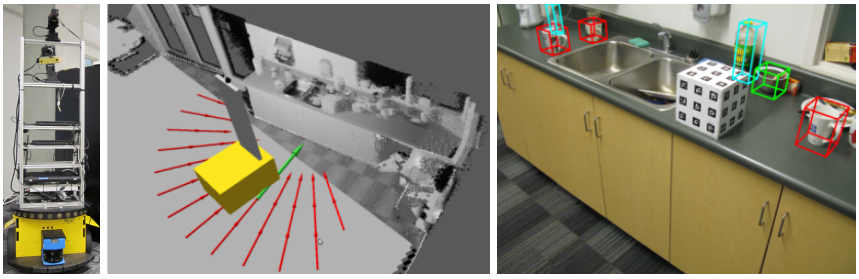


Fig. 1.1 An overview of the information contained in the UBC VRS dataset. Our robot collects a number of images of a scene. Geometric registration allows 3D object information to be projected into each image. Accurate 2D bounding box annotations are also provided.

During training and testing of robot recognition algorithms, the recorded data is provided to recognition algorithms by a simulator that mimics a robot's sensing and response to control input. We refer to this procedure as *simulation with real data*. Except for small limitations due to sampling discretizations, this perceptual experience is identical to the one a novel robot would experience in the same environments. This allows realistic evaluation of robot object recognition performance. Details on each stage in the process are provided in the remainder of this Section.

1.2.1 Robotic Data Collection

The sensor data that comprises the UBC VRS dataset was collected with the Curious George robot that is described in [16] and is shown in Figure 1.2(a). During data collection, the robot moves through a dense set of poses covering the space of possible visual experiences. We achieved this by planning a path consisting of three concentric circles. Along each circle, stop-points were located at an angular spacing of at most ten degrees (in some cases at a finer resolution). When the robot reached each stop point, it turned to face the center of the scene and it collected a single reading from each of its sensors. Figure 1.2(b) shows a sample path in one environment.



(a) (b) (c)

Sub-set	Scenes	Views	Instances	Boxes	Mugs	Bottles	Bowls
Training	19	453	184	4026	56/1405	42/1119	17/378
Validation	11	295	116	2701	33/711	33/839	17/346
Test	30	334	303	3466	85/935	57/589	64/681
UBC VRS	60	1082	603	10193	174/3051	132/2547	98/1405

(d)

Fig. 1.2 The UBC VRS Dataset. (a) The Curious George robot platform used for data collection. From top to bottom, the sensors include a tilting laser range-finder, digital camera, stereo camera and fixed laser range-finder. (b) A sample point cloud, and poses from the survey path followed by the robot. (c) A sample image with 3D wire-frames projected to display user-annotated ground truth volumes. (d) Summary statistics of the annotations available for the UBC VRS database. The final 3 columns represent the (unique instances / number of bounding boxes) that are present for the specified category.

In the ideal case, this data collection method ensures that real sensor readings are available from a viewpoint within 5 degrees of any pose requested by a simulator. However, constraints of our robot and the environments prevented a complete sampling. Factors such as building layout, uneven floors, and furniture obstacles caused the robot's navigation routines to skip some of the requested stop-points. Data from these skipped viewpoints is not available to recognition methods, which is also the case for real robotic systems exploring an environment. Recognition methods must therefore be robust to this realistic property of the dataset. Figure 1.2(d) displays the final number of images and scenes that were collected.

The Curious George robot has a variety of sensors suitable for object recognition. Images from the robot's high-resolution digital camera capable of 10 mega-pixel imaging are down-sampled to 1600 by 1200 pixel resolution and stored, to balance overall data size with sufficient resolution to capture objects in detail. A planar laser range-finder was tilted with a continuous periodic command to capture an entire 3D sweep of the scene from each viewpoint. The set of scans was then assembled to form a cloud comprised of roughly 500,000 individual points. Each point is represented with a 3D (X,Y,Z) position as well as an intensity value measured by the laser. During data collection, the relative positions of the robot's sensors were calibrated as often as possible. This involved estimating the transformation relating the camera to the laser with the so-called Laser-Camera Calibration Toolbox [21]. However, a moderate degree of calibration error remains a factor, as is the case for many commodity robotic platforms.

1.2.2 Geometric Registration

When a physical robot platform explores an environment, it has access to several forms of sensor feedback that can be used to determine its position. Also, it can actively control its position by issuing movement commands. In order to replicate this situation as closely as possible when performing recognition from our pre-recorded data, all information in the database is registered to a common base frame. First, the set of camera poses is registered using automatically detected fiducial marker points that correspond to known 3D target geometry to solve for the camera poses in a global frame. Then, the pre-calibrated relative sensor transformations (camera to X) are applied to globally register the remaining set of sensors. Using this common registration, both path information and robot control can be simulated, in combination with the real sensor data. This Section will describe the process for registering the camera poses in detail.

The cube-shaped target displayed in our example images (e.g. Figure 1.2(c)) is comprised of ARTag visual fiducial markers [10] and we have manufactured the cube target with precise 3D geometry. The ARTag library provides a marker detection scheme with virtually zero false positives that simultaneously localizes the corners of the fiducial patches in the image with sub-pixel accuracy. Each detected image location provides a 2D to 3D constraint on the extrinsic camera parameters (pose) using the typical pinhole camera projective equation:

$$\alpha \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = K[R|t] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{1.1}$$

where: x and y are the image coordinates of the detected corner pixel; K is the known intrinsic camera calibration containing the focal length, offsets and skew; R and t are the unknown rotation and translation which we seek; X, Y, and Z are the 3D coordinates of the corner point using the known layout of the target; and α represents projective scale. Numerous points are required to uniquely determine the camera pose, and our target provides between 36 and 108 visible corners, depending on the viewpoint. This yields a highly over-determined system. We estimate the solution using an approach similar to camera calibration methods such as [20], which involves making an initial guess using homography constraints (which exploits the known planarity of the target's faces), and then by refining the estimate using the Levenberg – Marquardt algorithm to minimize re-projection error.

We have validated this registration method on a number of test images by projecting known 3D points (e.g. a cube corner, or another point we have physically measured in 3D) into each of the images and manually observing the error in re-projection. The registration is typically accurate to within a pixel with the maximum error on the order of several pixels. Figure 1.3 illustrates the registration accuracy

(a) (b) (c)

Fig. 1.3 Example results of automated geometric image registration. Each column holds two views of the same scene. Our system uses the estimated camera positions relative to a global frame along with previous intrinsic calibration estimates to render a wire-frame of the extents of the calibration target (shown in red where colour is available) into each view. Accurate alignment of the wire-frame to image content indicates accurate registration.

in a set of example images. Registration information is stored with the raw sensory data and both are used during annotation and simulation of robot motion for testing.

1.2.3 Object Annotation

In order to evaluate the performance of recognition algorithms, a human has annotated each scene and image in the dataset. We seek to describe objects both in 3D in the common registration coordinate frame as well as in 2D in each image. Annotating this information is a time-consuming process, but we have leveraged the registration information described above to ease the manual burden. We provide the annotator with a software tool to triangulate a number of 2D object points to locate the 3D centroid, a set of controls to fine tune the object's orientation and scale in 3D, and a mechanism to refine an automatically initialized 2D bounding box for each object. We continue by providing more detail on both the 2D and 3D annotation procedure.

As mentioned previously, each image in our dataset has been accurately registered into a common coordinate frame. This allows projection of 3D information into each image, and it also permits triangulating a set of image points. The first step in our annotation process is for a human to mark a central and identifiable feature on an object in 3 or more images. We then solve for a 3D point that falls closest to the rays through each marked pixel. As described in Hartley *et al.* [12], this in-

volves finding the smallest singular vector of a matrix, A, formed by stacking rows that express constraints induced by the projection matrix and marked image points:

$$A \equiv \begin{bmatrix} xP_3 - P_1 \\ yP_3 - P_2 \end{bmatrix} \tag{1.2}$$

Here P is the three row by four column projection matrix combining extrinsic and intrinsic parameters: $P \equiv K[R|t]$ and underscore notation indicates selecting a particular 1-indexed row of the matrix. The result of triangulation is an estimated 3D center point for the object. Our annotation tool instantiates a 3D object region composed of a 3D centroid initialized to the triangulated point, a 3D scale initialized to be the mean size of the object category, and an azimuth angle (that is rotation around the *up* or Z axis) initialized to zero. The annotator is then able to refine each dimension, but we have found that, if the image points are specified accurately at the outset, there is little extra effort required beyond specifying the true object orientation. Upon approving of all properties of the 3D annotation, the annotator saves the object volume and this is recorded along with the sensor data and registration information to be available at test time.

Our annotators have also provided 2D annotations of objects in every image in the dataset. Our 2D annotations share the format used by Pascal and other recognition challenges. That is the bounding box of the object is drawn, with extents tight to the image content. The object's category type is recorded as well as additional meta-information such as that the instance may be *difficult*, in that it is an uncommon representative of the class (e.g. a toy coffee mug in the shape of a cartoon character is a difficult mug), or that the instance is *occluded* in the image. The previously created 3D annotations are leveraged to expedite the process of creating 2D annotations. Volumes are projected into every image in which they are visible, and a bounding box that encompasses the 3D corners is created. The annotator's task is then simply to refine the precise image locations and meta-information values, rather than having to create each bounding box. This saves significant effort and reduces the probability that an image region will be missed due to human error.

At this stage, the annotator also often makes small adjustments to the 2D bounding box to ensure that it is pixel-tight to the underlying image content. This hand-adjustment is needed because we project imprecise shape models (a box-shaped 3D volume, rather than the object's true shape), and to account for any small errors introduced by 3D to 2D projection. Once again, when the annotator is satisfied with the quality of the data, the 2D box and meta-information are saved to the database.

The code and tools of our labeling pipeline can be re-used for any series of moderately well-registered images (such as video sequences, well-calibrated vehicles possessing accurate inertial positioning and a camera, or sets of highly overlapping photographs). It has been made open-source to the community and is available online along with the dataset, as described below.

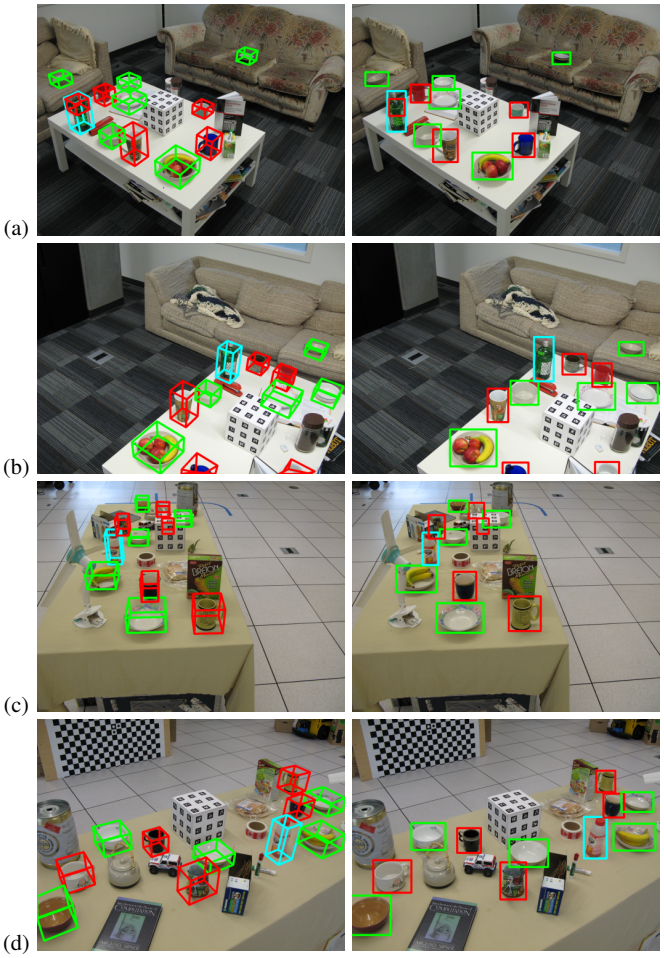


Fig. 1.4 Example annotations produced by a human. The left column shows 3D annotations projected onto the image (another verification of accurate registration) and the right column represents annotations that have been made directly on the 2D data, initialized by the projections. The first pair of rows, (a) and (b), are two views of the same scene, and the second pair of rows (c) and (d) are a second scene. The colours (where available) represent the object category with: bowls in green, mugs in red and bottles in blue.

1.2.4 Evaluation Protocols

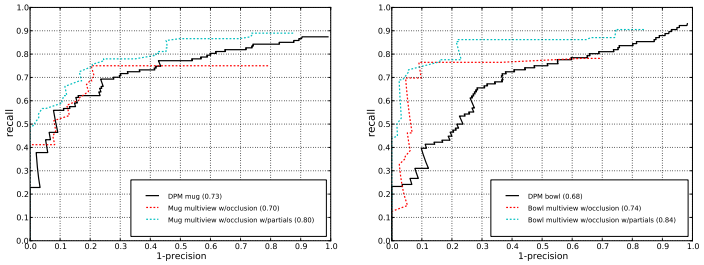
Our goal in collecting the UBC VRS dataset has been to facilitate scientific exploration of the robot recognition problem. That requires researchers to be able to compare the results of different methods evaluated on the same task. To that end, we describe a set of protocols that leverage the information provided in the dataset to simulate a variety of tasks. In each case, we also outline the performance metric

that is appropriate for benchmarking and comparison, using established guidelines where applicable. We focus on *simulate with real data* protocols since we believe this is crucial to addressing the complicated set of challenges facing a system that searches for objects in unstructured environments. This Section will describe the protocols in detail:

- *Passive single-view recognition:* means that each image is treated independently, as is common in the Computer Vision field. No registration or path information is available. In this case algorithms can localize objects with 2D bounding boxes in images, or by estimating 3D objects from 2D imagery and point clouds. The widely accepted metric for evaluation of such methods is precision and recall (PR) curves and the average precision (AP) statistic. Such curves are produced by varying a confidence threshold for the recognition method, and comparing which of the hypothesized 2D bounding boxes correctly overlaps an annotated (ground truth) object region. For each threshold, the ratio of true positives to total annotated objects is known as recall, while the ratio of true positives to number of hypothesized objects is precision. Average precision summarizes performance across all possible thresholds. Perfect performance on the task would give all of precision, recall, and average precision equal to 1.0.
- *Passive multi-view recognition:* provides the recognizer with data from a series of poses with variable length, N, from a path chosen by the simulator at random in each trial. Position information is provided to the recognizer so that evidence can be integrated spatially. This task captures the unique ability of a robot to move through the scene and gather data. Results may be reported as 3D object poses defined in the global registration frame, or as individual sets of 2D objects in each image. To standardize comparison, and to utilize the most accurate source of ground truth in the dataset, the primary evaluation should again be PR curves and AP statistics based on evaluation of 2D hypotheses. Where methods report 3D object poses, these should be projected to form 2D bounding boxes in each image. Our toolbox provides this functionality to aid fair evaluation.
- *Active multi-view recognition:* extends the passive protocol by requiring the recognition algorithm to provide a control input which simulates the robot's motion to a new pose and alters the image sequence accordingly. In general, a robot could take an unbounded number of steps through an environment before estimating the objects present, but we must standardize on specific path lengths for comparison with the passive multi-view protocol. Therefore, evaluation should be done after N steps are chosen by the system and the data is analyzed. All other evaluation details should remain the same as described in the previous item.

1.3 Experiments

At the time of submission of this abstract, several researchers have attempted to perform a subset of the robot recognition tasks possible with the UBC VRS dataset [17, 13, 18, 19]. This section will describe the results of these previous methods



(a) (b)

Fig. 1.5 Results of several versions of the the passive multi-view recognition method from [19] are compared against a state-of-the-art passive single-view method from [9] (labeled DPM) which has won the Pascal VOC challenge for several years. The summary statistic is Average Precision.

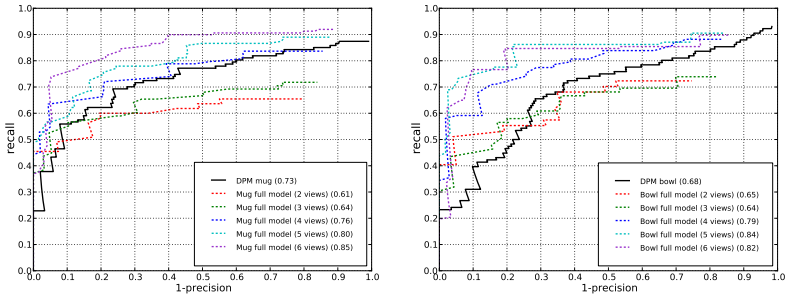
along with a brief description of the methods achieving the highest performance at the time of this publication. Further improvements will be recorded on the dataset's website.

1.3.1 Passive Single Viewpoint Recognition

Meger *et al.* [19] report an average precision of 0.73 for mugs and 0.68 was achieved across the UBC VRS test set by the method of Felzenszwalb *et al.* known as the *Deformable-Parts Model* DPM. The DPM method is based on learning a latent-variable Support-Vector Machine (LVSVM) [3] classifier over an image feature similar to the well-known Histogram of Oriented Gradients (HOG) [4]. The authors attempted to train the DPM model on a variety of training data sets, and found maximum performance by selecting positive examples from both ImageNet [5] as well as from the UBC VRS training set, and by including a large number of relevant negative images from a variety of sources. Single-view recognition is not the focus of our dataset, but it is informative to observe whether an improvement can be made by fusing information from multiple viewpoints, and thus this DPM result serves as a baseline for comparison with other methods.

1.3.2 Passive Multiple Viewpoint Recognition

The highest passive multiple viewpoint recognition performance reported in previous work was obtained by [19] and we present the results of that paper here as a current benchmark. Their solution leveraged strong single-view DPM hypotheses in each image, and lifted this 2D appearance information to allow explicit per-object occlusion inference and the use of part-based detectors to improve accuracy on the



(a) (b)

Fig. 1.6 Results of passive multi-view recognition over various numbers of views per scene, from [19] are compared against a state-of-the-art passive single-view method from [9] (labeled DPM) which has won the Pascal VOC challenge for several years. The summary statistic is Average Precision.

large number of partially occluded object instances found in the dataset. Figures 1.5 and 1.6 illustrate the performance of this method on the UBC VRS test set. The first figure demonstrates results of the method when the simulator randomly provides images from five viewpoints per scene. By using multiple viewpoints, the method is able to achieve higher average precision than the single-view baseline provided by DPM. Also, occlusion reasoning is shown to give an improvement in the performance of the method, and additionally including part-based appearance information leads to the highest performance overall.

Figure 1.6 expands upon the previous result by examining the performance of the method as various numbers of viewpoints are made available for each scene. In all cases, the system achieves more recall in the high-precision region of the curves, but in some cases, the overall performance of the multi-view method is worse than the single-view baseline. This was explained by the difficulty in recovering 3D information from the weak geometric cues available in a small number of views that are often from a wide-baseline. However, it remains to be seen if future approaches will be able to achieve better performance from only two or three views of a scene. In all cases, the clear trend is that the method performs better as more views become available.

1.3.3 Active Multiple Viewpoint Recognition

As of the submission of this abstract, only a very preliminary attempt has been made on the active multi-view recognition task on a small portion of the dataset, by [17]. Active recognition (i.e. explicitly selecting the next best viewpoint at each step) is easily done with the UBC VRS dataset and is an area for future work.

1.4 Experimental Insights

While careful collection and annotation of a dataset with sufficient scale for meaningful evaluation is a large effort, the resulting repeatable evaluation will hopefully be of value to the robotics community[1]. Beyond data, a key contribution of our method is the labeling and evaluation pipelines, and the tools related to these can extend to a variety of additional data sources. For example, we have already succeeded in using the same tools to annotate and evaluate our approaches using Kinect data that was registered without the use of our fiducial marker (i.e. using the software of Endres *et al.* [6]) as well as outdoor data collected by an automobile with a highly accurate inertial measurement unit.

Pascal VOC has encouraged various authors to borrow and improve upon the best techniques from the winners of the previous years. A number of authors have already obtained the UBC VRS and are currently beginning to develop new solutions. Ideally, this will lead to additional performance improvements being published in coming years by a variety of authors. The data has been collected by a physical robotic platform along with modern sensors, control, and calibration. So, pursuit of such improvement is likely to provide direct benefits to the ability of many robots to perceive objects, in many environments.

Acknowledgements The authors thank Scott Helmer and Marius Muja for their assistance with the early development of tools and collection of the first portion of data that lead to the final UBC VRS dataset. This research was funded, in part by the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

1. Website: http://www.semantic-robot-vision-challenge.org/.
2. Website: http://solutionsinperception.org/index.html/.
3. S. Andrews, I. Tsochantaridis, , and T. Hofmann. Support vector machines for multiple-instance learning. In *Advances in Neural Information Processing Systems*, 2003.
4. N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 886 – 893, San Diego, USA, June 2005.
5. J. Deng, K. Li, M. Do, H. Su, and L. Fei-Fei. Construction and Analysis of a Large Scale Image Ontology. Vision Sciences Society, 2009.
6. F. Endres, M. Hess, N. Engelhard, J. Sturm, D. Cremers, and W. Burgard. An evaluation of the rgb-d slam system. In *Proceedings of ICRA*, 2012.
7. M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2011 (VOC2011) Results. In *Website*, 2011.
8. L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshop on Generative-Model Based Vision*, 2004.

[1] The data and code is are both completely open, at http://www.cs.ubc.ca/labs/lci/vrs/index.html

9. P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32, 2010.
10. M. Fiala. Artag, a fiducial marker system using digital techniques. In *CVPR'05*, volume 1, pages 590 – 596, 2005.
11. G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical Report 7694, California Institute of Technology, 2007.
12. R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049, 2000.
13. Scott Helmer, David Meger, Marius Muja, James J. Little, and David G. Lowe. Multiple viewpoint recognition and localization. In *Proceedings of the Asian Computer Vision Conference*, 2010.
14. A. Janoch, S. Karayev, Y. Jia, J.T. Barron, M. Fritz, K. Saenko, and T. Darrell. A category-level 3-d object dataset: Putting the kinect to work. In *Proceedings of the ICCV Workshop on Consumer Depth Cameras in Computer Vision*, 2011.
15. Kevin Lai, Liefeng Bo, Xiaofeng Ren, and Dieter Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *IEEE International Conference on on Robotics and Automation*, 2011.
16. David Meger, Per-Erik Forssén, Kevin Lai, Scott Helmer, Sancho McCann, Tristram Southey, Matthew Baumann, James J. Little, David G. Lowe, and Bruce Dow. Curious george: An attentive semantic robot. *Robotics and Autonomous Systems Journal Special Issue on From Sensors to Human Spatial Concepts*, 56(6):503–511, 2008.
17. David Meger, Ankur Gupta, and James J. Little. Viewpoint detection models for sequential embodied object category recognition. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 2010.
18. David Meger and James J. Little. Mobile 3d object detection in clutter. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, San Fransisco, United States, 2011.
19. David Meger, Christian Wojek, Bernt Schiele, and James J. Little. Explicit occlusion reasoning for 3d object detection. In *Proceedings of the 22nd British Machine Vision Conference (BMVC)*, 2011.
20. R. Y. Tsai and R. K. Lenz. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, pages 323–344, 1987.
21. Ranjith Unnikrishnan and Martial Hebert. Fast extrinsic calibration of a laser rangefinder to a camera. Technical Report CMU-RI-TR-05-09, Robotics Institute, Pittsburgh, PA, July 2005.