Occlusion Alleviation through Motion Using a Mobile Robot

Duc Fehr, William J. Beksi, Dimitris Zermas and Nikolaos Papanikolopoulos
{fehr, beksi, dzermas, npapas}@cs.umn.edu
Department of Computer Science and Engineering
University of Minnesota
Minneapolis, MN 55455

Abstract—**Object segmentation and classification is an important and difficult task in robotic vision. The task is complicated even further when the different objects are partially or completely occluded. Allowing a robot to take measurements from varying points of view can help in alleviating or completely removing occlusions. A robot equipped with an RGB-D sensor has the capability of searching for new and better points of view to facilitate object recognition. In this work, a motion control algorithm is designed and implemented on a mobile robot to facilitate object classification in RGB-D data of clustered objects.**

I. INTRODUCTION

On factory assembly lines and in warehouse settings, robots routinely need to pick out objects, identify them, and bring them to the proper stations. To perform these tasks, object segmentation and classification is of crucial importance. Detecting, segmenting, and classifying objects in images has been intensely studied. The problem is more complicated when the objects are partially or completely occluded given the current viewing angle. In the past, various methods have been proposed to deal with these problems. However, many of these techniques do not utilize the advantage of a mobile robot with a suitable sensor that can move about the environment in order to alleviate occlusions.

In this paper, we introduce a system that uses a control strategy to circumnavigate objects in order to capture a view of the objects from several different angles. These different points of view help the machine learning classification algorithm in labeling the objects. The sensor used with the mobile robot, the Microvision robot (Fig. 1), is an RGB-D camera. The RGB-D camera not only provides a regular image, but also a depth map which can be merged into a "colored" point cloud. The point clouds are then used to perform object segmentation and classification.

Due to the recent introduction of affordable RGB-D cameras, research in the area of 3D point cloud processing is thriving. This research involves the object segmentation and feature processing steps. In addition, several different classification methods have been introduced and work on different descriptors is ongoing. Despite recent advances in the field, the occlusion problem is still present and there is a definite advantage of having a robot equipped with an RGB-D sensor compared to a camera that is statically placed.

This paper presents the possibilities and advantages that a mobile robot provides over a spatially fixed sensor. Mobile robots are ideal sensor carrying platforms. Their ability to
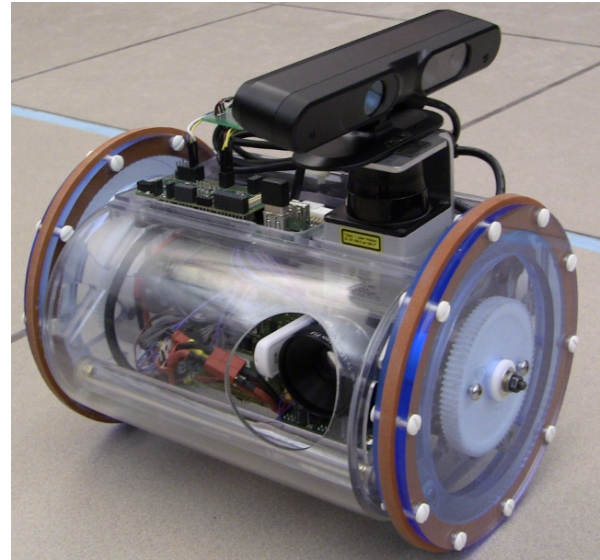

Fig. 1: The Microvision robot equipped with an RGB-D camera.

move about the environment instead of remaining in a fixed location (e.g., industrial robots) allows them to dynamically sense and report on their surroundings. As research expands in the area of robotics, we can expect mobile robots to increasingly take on the role of performing object recognition in various scenarios.

The remainder of this paper is organized as follows. Presented in Section II is related work that provides solutions to occlusions through mobile sensors. Section III describes the motion scheme used to intelligently circumnavigate the different objects. The covariance based classification is described in Section IV. Section V discusses the Microvision robot used in this paper followed by experimental results (Section VI). The paper concludes in the last section with an outlook on future work.

II. RELATED WORK

This paper associates with several different aspects of computer vision and machine learning. Two of its main topics are the following: occlusion alleviation and RGB-D data classification. Below we discuss past and current work in these areas.

Occlusions always present difficulties when performing

object recognition. In the areas of vehicle and pedestrian tracking, much research has been done regarding the alleviation of occlusions. Although these areas address a slightly different problem, they nevertheless offer important findings in the pursuit of a solution. Pang *et al.* [1] fit models around different vehicles and uses anomalies in the model size ratios to detect vehicle occlusions. This is then followed by a partitioning of the model to separate the foreground and the occluded object. Ghasemi and Safabakhsh [2] use a Kalman filter to track vehicles and resolve occurring occlusions. Xing *et al.* [3] try to associate tracklets of partially and heavily occluded pedestrians using particle filters with observer selection. When tracking generic objects in stationary scenes with fixed sensors, it is difficult to overcome heavy occlusions.

By observing the movement of objects, Wang *et al.* [4] propose a novel approach by modeling the occluder rather than the occluded object and scanning the area around the occluder until the object of interest reappears. Guha *et al.* [5] show that only 7 states of occlusions are possible and formulate a systematic approach which they call Oc-7. Alternatively, occlusions can be alleviated by adding movement to the sensor itself. Blaer and Allen [6] and Maver and Bajcsy [7] have implemented a next best view approach by utilizing a mobile robot in order to acquire better views of occluded objects. Radmard *et al.* [8] use a robotic manipulator in order to overcome obstacles and reach the occluded object.

One of the most important works relating to RGB-D data classification has been performed by Lai *et al.* [9], in which the authors introduce a database for testing purposes. Many different algorithms have been tested against this database with more or less success. For instance the same authors use sparsity techniques to classify objects in [10]. Kernel methods for recognition have been used by Bo *et al.* [11]. The authors refine this method by adding a hierarchical model [12]. Dictionary learning to find good features is used by Blum *et al.* [13].

Classification databases usually provide clean, occlusion free, data. Under occlusions, the problem becomes more complex. Several approaches for the classification problem have been proposed using static sensors and RGB-D data. For instance, in [14], Tombari *et al.* introduce a Hough voting technique that helps with the problem of recognition through occlusion. Merchan *et al.* [15] have developed the Depth Gradient Image Based on Silhouette representation algorithm to deal with the object classification problem under occlusions.

III. MOTION CONTROL SCHEME

The motion control scheme involves the robot navigating around a cluster of objects for the purpose of obtaining different viewpoints of the objects within the cluster. Previous work, [16] and [17], uses a similar scheme in connection with a laser range finder. Data from the RGB-D camera is used to compute the convex hull of the detected objects and an aiming point used for velocity controls. The RGB-D camera is positioned along the y_R axis of the robot. This
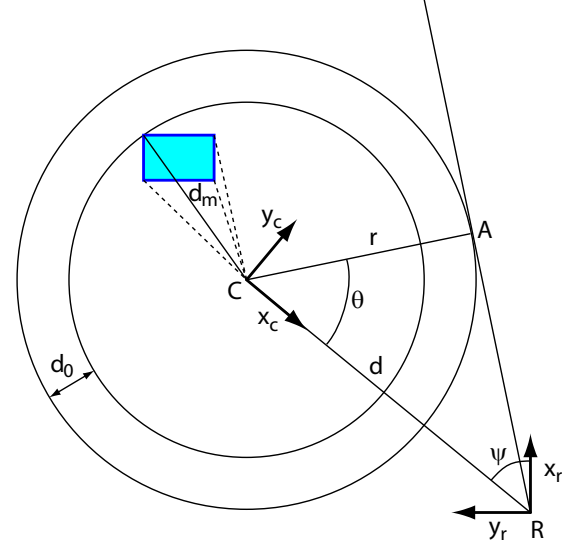

Fig. 2: This figure shows the definition of the aiming point A. R is the center of the robot reference frame, and C the centroid of all the detected objects. The blue square corresponds to one detected object.

pose enables the robot to smoothly circle the objects without having to rotate the camera to keep the objects of interest in the field of view.

First, the points on the objects are projected onto the ground plane. Then their centroid and convex hull in the same plane are computed. Finally, the aiming point A is derived, which is depicted in Fig. 2 along with the different points and angles used in the computation. R is the center of the robot reference frame. C is the centroid of all the detected objects. A is the intersection of the circle centered at C of radius r with the line containing R tangential to that circle. The blue square represents one of the detected objects that is the farthest away from the centroid. This distance defines d_m. d_0 sets the offset distance at which the objects are to be circled. From there, the aiming point A can be defined in the robot centric coordinate frame as follows:

$$^C A = \begin{pmatrix} r\cos\theta \\ r\sin\theta \end{pmatrix} \tag{1}$$

$$^R A = {}^R C + {}^R_C T(\psi')\ {}^C A \tag{2}$$

$$^R A = \begin{pmatrix} x_c \\ y_c \end{pmatrix} + \begin{pmatrix} \cos\psi' & -\sin\psi' \\ \sin\psi' & \cos\psi' \end{pmatrix} \begin{pmatrix} r\cos\theta \\ r\sin\theta \end{pmatrix}, \tag{3}$$

where

$$r = \|CA\| \tag{4}$$
$$= d_m + d_0 \tag{5}$$
$$d = \|CR\| \tag{6}$$
$$\psi = \tan^{-1}(C_y/C_x) \tag{7}$$
$$\psi' = \psi + \pi \tag{8}$$

$$\theta = \widehat{RCA} \tag{9}$$

$$\triangle(CAR) = \perp \tag{10}$$

$$\cos\theta = \frac{r}{d}. \tag{11}$$

Eq. (11) shows that if the robot is too close to the centroid of the objects, i.e., $r > d$, θ is not defined. In this case d is set to $d = r + r_0$, with r_0 an offset, allowing A to still be available.

The linear and rotational velocities, V and ω respectively, are then set proportionally to the distance and angle of the aiming point:

$$V = K_v \|RA\| \tag{12}$$

$$\omega = K_\omega \tan^{-1}(A_y/A_x), \tag{13}$$

where K_v and K_ω are the gain coefficients.

IV. OBJECT CLASSIFICATION

Covariance based descriptors are used for object classification. They represent a new paradigm for the classification of point clouds and were introduced by Tuzel *et al.* [18] and Porikli *et al.* [19] for people tracking in the area of image processing. They have shown exceptional results not only for people tracking, but also for other domains such as face recognition. Additionally, Pang *et al.* [20] describe covariances descriptors built on Gabor filters that perform very well.

Classification using covariance based descriptors on 3D point cloud data has shown promising results in [21]. These descriptors are being developed further in [22]. In the current work, the point clouds provide nine different features for each point producing the following feature vector:

$$f = [x, y, z, R, G, B, n_x, n_y, n_z] \tag{14}$$

The features used are the Cartesian coordinates (x, y, z), the color channel values (R, G, B), and the normal coordinates (n_x, n_y, n_z) at the specific point.

From the feature vector f of each point, the covariance C of an object can be computed:

$$C = \frac{1}{N-1} \sum_{i=1}^{N} (f_i - \mu_f)(f_i - \mu_f)^T \tag{15}$$

where N is the number of points in the object and i the point's index in the objects list. μ_f is the mean of the considered feature vectors.

These covariances characterize the objects and are the descriptors on which the classification is done. The classification uses a support vector machine (SVM, [23]) with a radial basis function, $\exp\left(-\gamma\, d^2\left(C_1, C_2\right)\right)$, on the distance d between the covariances C_1 and C_2. Although the geodesic distance developed by Förstner and Moonen [24] produces the exact distance between two covariance matrices, its determination is computationally expensive. An estimate of the distance between covariances, the log Euclidean distance, has been developed by Arsigny *et al.* [25], which can be computed much faster. It is defined as the Frobenius norm of the difference of the matrix logarithms of the covariances.

$$d\left(C_1, C_2\right) = \|\log\left(C_1\right) - \log\left(C_2\right)\|_F \tag{16}$$

Since the matrix logarithm computations are decoupled, they can be computed separately allowing for faster computation.

V. ROBOT DESCRIPTION

Developed at the University of Minnesota's Center for Distributed Robotics, the Microvision is a versatile robotics platform [26]. The robot is equipped with a scanning laser range finder, RGB and RGB-D cameras, and audio stream capture ability. Its on-board computational power and sensor payload give the robot the capability to run a number of applications under different scenarios. Within the laboratory, the Microvision is deployed for experimental research in robotics and computer vision.

Mounted on top of the Microvision is the Asus Xtion Pro Live. The Xtion is a motion sensing device equipped with an RGB-D camera and a pair of microphones. The depth camera has a range of 0.8 m to 3.5 m. It has viewing angles of $45°$ in the vertical direction, $58°$ in the horizontal direction, and $70°$ in the diagonal direction. The Xtion is positioned at a $90°$ offset relative to the base of the robot.

The Microvision runs the Robot Operating System (ROS) [27]. To facilitate occlusion alleviation, a ROS node has been developed which implements the motion control scheme in Section III. This node performs three main services. First, it implements a callback function for receiving point cloud data from the RGB-D camera. Second, the node processes the point cloud data and computes a new aiming point as described in the motion control scheme. Lastly, the node sends linear and angular velocities to the Microvision ROS node based on the location of the computed aiming point.

The Microvision ROS node functions as the driver for the robot and provides control to the robot's wheels and tail. The node subscribes to linear and angular velocity messages published by the motion control node. Throughout this process, point clouds are saved to a disk for offline processing. The point cloud processing pipeline is currently being optimized to perform classification tasks in real time.

VI. EXPERIMENTAL RESULTS

A. Setup

The different objects used in this experiment are given in Fig. 3. The top row ((a) - (f)) shows a full image of the different objects. The middle row ((g) - (l)) shows a view from the robot and the bottom row ((m) - (r)) shows the color associated with the different objects.

The robot is set up to autonomously navigate around the different objects following the strategy described in Section III. Fig. 4 shows the setup of the experiment.

Fig. 5 shows the robot's computation of the aiming point, which is shown in cyan. The robot is represented by a blue dot and the objects are displayed in green. The centroid of the objects is in red and the aiming circle is drawn in yellow.

| (a) Box | (b) Coffee can | (c) Water heater | (d) Paper roll | (e) Shoe | (f) Shuttle |

| (g) Box | (h) Coffee can | (i) Water heater | (j) Paper roll | (k) Shoe | (l) Shuttle |

| (m) Box | (n) Coffee can | (o) Water heater | (p) Paper roll | (q) Shoe | (r) Shuttle |

Fig. 3: These figures show the different objects used in the experiment. The top row gives an RGB view of the objects, the middle row depicts the robot's view of the object, and the bottom row gives the color coding corresponding to each object.
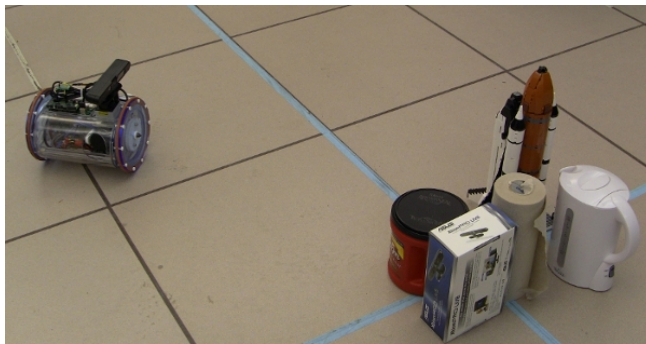

Fig. 4: Robot and object setup.

B. Classification

Figs. 6 to 8 provide results for a couple of experimental runs. While circling, the robot is able to capture and recognize the different objects. When there is no occlusion (Fig. 6), the segmentation of the objects is relatively simple and the covariance descriptor provides excellent results. In some cases where there is slight overlap (Figs. 7 and 8), the classification still provides good results.

A difficulty in this approach is encountered when the segmentation merges parts that should not be merged, or separates parts of the point cloud that should be together. Fig. 9 shows an example of this problem, which is from the same run as Fig. 8. In this figure, the coffee can and paper roll clusters get merged and as a consequence they are wrongly classified. A better and more robust segmentation scheme is necessary to address this issue.

When there is too much occlusion the classification breaks down, however the circling robot can provide another point of view from which it is possible to correctly classify. Fig. 10 shows an instance in which occlusion leads to misclassification. This view is taken from the same run as in Fig. 8. Only the shuttle's nose is in the robot's field of view. If the robot can only see parts of an object, then it can be problematic to categorize the object since the classifier has been trained on full, non-occluded views. As the robot moves around the cluster, the field of view changes to allow for better classification.

VII. CONCLUSION AND FUTURE WORK

A circling robot with point cloud capture ability provides a robust solution to the classification of occluded objects. If a certain point of view does not provide a good angle, other views readily become available after the robot adjusts
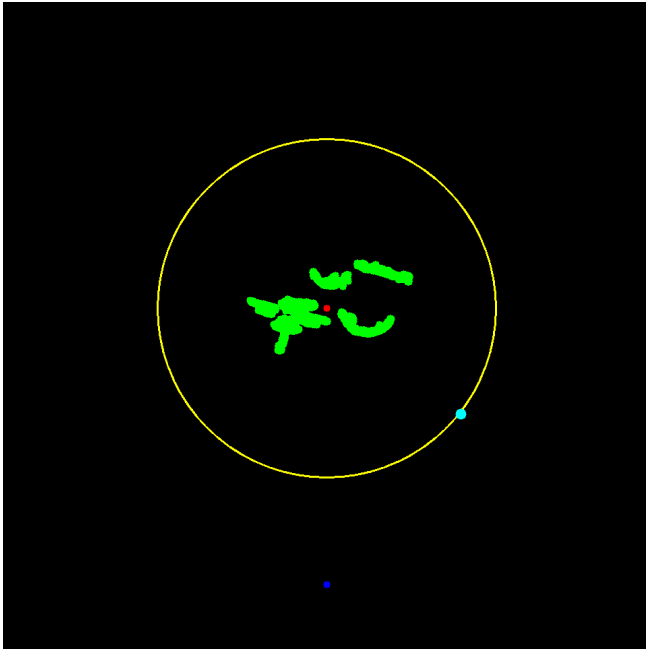
Fig. 5: Aim (cyan) computed by the robot (blue) computing the aiming circle (yellow) from the objects (green) and their centroid (red).
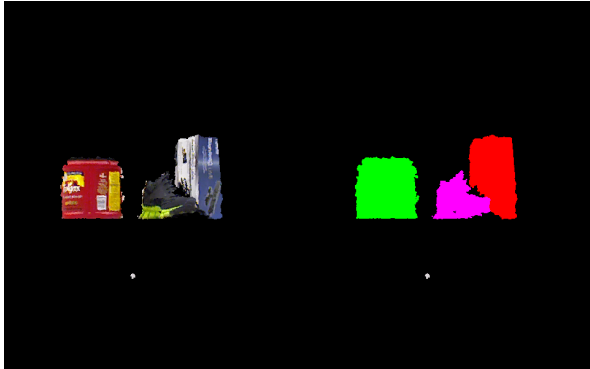

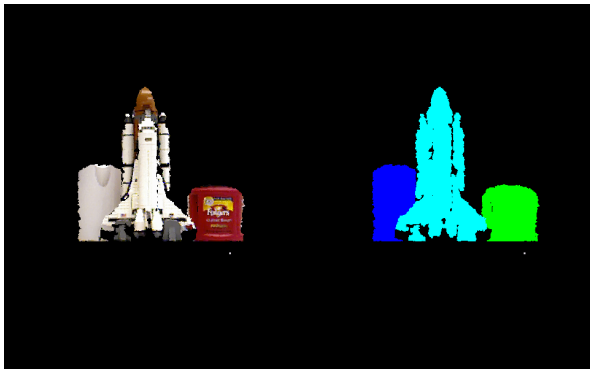Fig. 6: Result of one classification run with the objects' color code given in Fig. 3.

Fig. 7: Result of one classification run with the objects' color code given in Fig. 3.
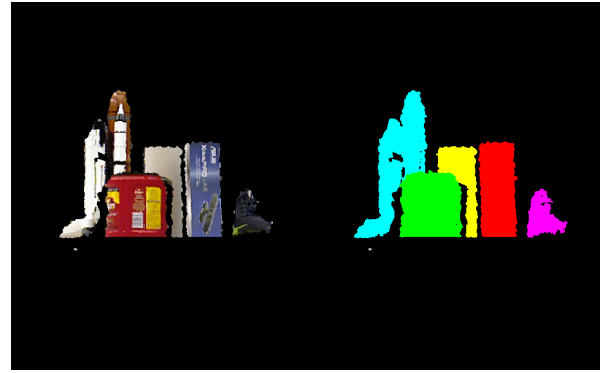

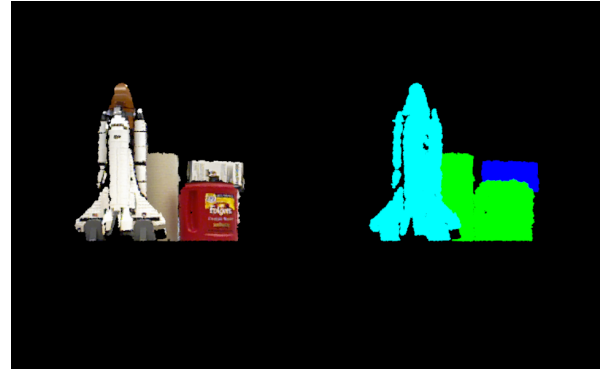Fig. 8: Result of one classification run with the objects' color code given in Fig. 3.

Fig. 9: This figure shows an instance where poor segmentation leads to misclassification. The coffee can and paper roll clusters are merged.

its position. These experiments have shown that a robot, equipped with an RGB-D camera and employing a motion control algorithm, can help alleviate problems that occlusions present for classification. This work also shows the potential of using a mobile robot for performing real-time object recognition within its surrounding environment.

Future work includes the development of a more dedicated motion control scheme. Other than an autonomous circum-navigating algorithm, a path planning strategy that steers the robot towards or around areas of occlusions is under development. The point density of a cluster can be used to drive the robot to different positions with better points of view [28], thus facilitating the object classification process.

As discussed in the previous section, a more robust segmentation scheme is necessary and is currently being worked on. Improved segmentation will naturally lead to improved classification. The merging and splitting of clusters within a point cloud remains an important and non-trivial problem.

Finally, the classification descriptor itself can be improved. For example, using a larger feature vector that contains more than only nine features is easily envisioned. However, a trade off needs to be found between the computation speed of the features and the additional performance provided. Furthermore, the performance of these improvements needs to be tested in a similar, real world scenario, as presented in
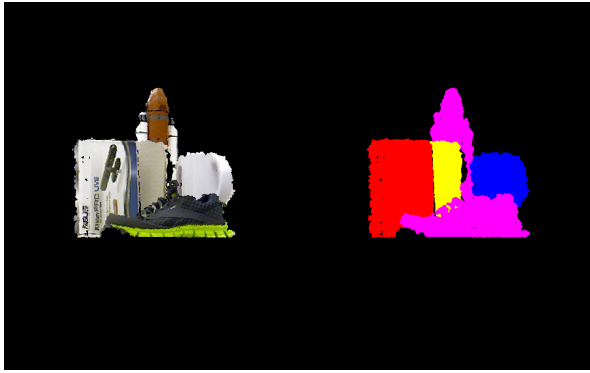
Fig. 10: This figure shows an instance where occlusion leads to misclassification. Most of the shuttle is occluded which leads to that segment being classified as part of the shoe.

this paper.

REFERENCES

[1] C. Pang, W. Lam, and N. H. C. Yung, "A novel method for resolving vehicle occlusion in a monocular traffic-image sequence," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, no. 3, pp. 129–141, 2004.

[2] A. Ghasemi and R. Safabakhsh, "A real-time multiple vehicle classification and tracking system with occlusion handling," in *Proceedings of the 2012 IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*, 2012, pp. 109–115.

[3] J. Xing, H. Ai, and S. Lao, "Multi-object tracking through occlusions by local tracklets filtering and global tracklets association with detection responses," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 1200–1207.

[4] P. Wang, W. Li, W. Zhu, and H. Qiao, "Object tracking with serious occlusion based on occluder modeling," in *International Conference on Mechatronics and Automation (ICMA)*, 2012, pp. 1960–1965.

[5] P. Guha, A. Mukerjee, and V. Subramanian, "Formulation, detection and application of occlusion states (oc-7) in the context of multiple object tracking," in *8th IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS)*, 2011, pp. 191–196.

[6] P. Blaer and P. Allen, "View planning and automated data acquisition for three-dimensional modeling of complex sites," *Journal of Field Robotics*, vol. 26, no. 11-12, pp. 865–891, 2009.

[7] J. Maver and R. Bajcsy, "Occlusions as a guide for planning the next view," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 5, pp. 417–433, 1993.

[8] S. Radmard, D. Meger, E. Croft, and J. Little, "Overcoming occlusions in eye-in-hand visual search," in *American Control Conference (ACC)*, 2012, pp. 4102–4107.

[9] K. Lai, L. Bo, X. Ren, and D. Fox, "A large-scale hierarchical multi-view RGB-D object dataset," in *Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 1817–1824.

[10] ——, "Sparse distance learning for object recognition combining rgb and depth information," in *Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 4007–4013.

[11] L. Bo, X. Ren, and D. Fox, "Depth kernel descriptors for object recognition," in *Proceedings of the 2011 IEEE Conference on Intelligent Robots and Systems (IROS)*, 2011, pp. 821–826.

[12] ——, "Unsupervised feature learning for RGB-D based object recognition," *ISER*, 2012.

[13] M. Blum, J. Springenberg, J. Wulfing, and M. Riedmiller, "A learned feature descriptor for object recognition in rgb-d data," in *Proceedings of the 2012 IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 1298–1303.

[14] F. Tombari and L. Di Stefano, "Object recognition in 3D scenes with occlusions and clutter by hough voting," in *Proceedings of the Fourth Pacific-Rim Symposium onImage and Video Technology (PSIVT)*, 2010, pp. 349–355.

[15] P. Merchan, A. Adan, and S. Salamanca, "Identification and pose under severe occlusion in range images," in *Proceedings of the 19th International Conference on Pattern Recognition (ICPR)*, 2008, pp. 1–4.

[16] D. Fehr and N. Papanikolopoulos, "Using a laser range finder mounted on a microvision robot to estimate environmental parameters," in *SPIE Defense, Security, and Sensing*. International Society for Optics and Photonics, 2009, p. 733211.

[17] P. Tokekar, V. Bhatawadekar, D. Fehr, and N. Papanikolopoulos, "Experiments in object reconstruction using a robot-mounted laser range-finder," in *17th Mediterranean Conference on Control and Automation, MED'09*. IEEE, 2009, pp. 946–951.

[18] O. Tuzel, F. Porikli, and P. Meer, "Region covariance: A fast descriptor for detection and classification," in *Computer Vision ECCV*, ser. Lecture Notes in Computer Science, A. Leonardis, H. Bischof, and A. Pinz, Eds. Springer Berlin / Heidelberg, 2006, vol. 3952, pp. 589–600.

[19] F. Porikli, O. Tuzel, and P. Meer, "Covariance tracking using model update based on lie algebra," in *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2006, pp. 728–735.

[20] Y. Pang, Y. Yuan, and X. Li, "Gabor-based region covariance matrices for face recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 7, pp. 989–993, 2008.

[21] D. Fehr, A. Cherian, R. Sivalingam, S. Nickolay, V. Morellas, and N. Papanikolopoulos, "Compact covariance descriptors in 3D point clouds for object recognition," in *Proceedings of the 2012 IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 1793–1798.

[22] D. Fehr, W. J. Beksi, D. Zermas, and N. Papanikolopoulos, "RGB-D object classification using covariance descriptors," *Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[23] C.-C. Chang and C.-J. Lin, "Libsvm: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 27:1–27:27, 2011.

[24] W. Forstner and B. Moonen, "A metric for covariance matrices," *Qua vadis geodesia*, pp. 113–128, 1999.

[25] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache, "Log-Euclidean metrics for fast and simple calculus on diffusion tensors," *Magnetic Resonance in Medicine*, vol. 56, no. 2, pp. 411–421, 2006.

[26] W. J. Beksi, K. Choi, D. Canelon, and N. Papanikolopoulos, "The microvision robot and its capabilities," *submitted to the 2014 IEEE Conference on Intelligent Robots and Systems (IROS)*, 2014.

[27] ROS. [Online]. Available: http://www.ros.org

[28] V. Bhatawadekar, D. Fehr, V. Morellas, and N. Papanikolopoulos, "A dynamic sensor placement algorithm for dense sampling," in *Proceedings of the 14th International Conference on Information Fusion (FUSION)*, 2011, pp. 1–7.