Object Classification Using Dictionary Learning and RGB-D Covariance Descriptors

William J. Beksi and Nikolaos Papanikolopoulos

***Abstract*— In this paper, we introduce a dictionary learning framework using RGB-D covariance descriptors on point cloud data for performing object classification. Dictionary learning in combination with RGB-D covariance descriptors provides a compact and flexible description of point cloud data. Furthermore, the proposed framework is ideal for updating and sharing dictionaries among robots in a decentralized or cloud network. This work demonstrates the increased performance of 3D object classification utilizing covariance descriptors and dictionary learning over previous results with experiments performed on a publicly available RGB-D database.**

I. Introduction

The ability of a machine to recognize objects is a fundamental problem and major area of research within robotics and computer vision. Although significant progress has been made in object recognition through a variety of algorithms in the last decade, the classification of common items in typical scenes is still an open research problem. The challenge of classification lies in the large intra-class variance due to the high dimensional image space which changes with each new object viewpoint, and the fact that objects may be deformable. The function of a system that performs object classification is to extract meaningful representations (features) from high dimensional data provided by images, videos, and 3D point clouds.

Many robots make use of visual information to interpret their surrounding environment and the reliable classification of objects can assist in this interpretation. In recent years, the availability of low-cost RGB-D sensors [1] which are able to provide synchronized color and depth images, have given robots the ability to utilize dense 3D point cloud data for visual perception. Although the availability of RGB-D sensors has led to richer datasets, as to which feature descriptors and machine learning techniques can best be used to increase the performance of object classification remains undetermined.

Many state of the art descriptors are based on extracting local features around points of interest in 3D data. The most prominent methods make use of signatures of histograms such as SHOT [2] and FPFH [3]. However, by only using the local geometry around a point, these methods discard other available information. Therefore, researchers have incorporated additional knowledge such as texture (CSHOT) [4] and viewpoint (VFH) [5] into their feature descriptors. In this work, we construct a feature descriptor that is composed of both geometric and color information computed over an entire object of interest.

The RGB-D covariance descriptor [6], [7] is a low dimensional 3D feature descriptor that can be computed very quickly. Different combinations of discriminative features can be used in the descriptor allowing for high flexibility. Moreover, the descriptor is extremely compact; an entire point cloud is represented by a single positive definite matrix. Our covariance descriptors are composed of both shape features (normals, principal curvatures, Gaussian curvature) and visual features (color, gradient, depth, etc.). We use these covariance descriptors in conjunction with sparse coding [8] to learn multiple basis sets, i.e. dictionaries, on the dataset.

In the area of performing object classification among groups of robots capable of capturing RGB-D point cloud data, little work has been done. Processing and classifying point cloud data is computationally demanding. Robots may not have the necessary on-board resources to perform this task, however computations and access to classifier models can be provided by a remote computing infrastructure or by other robots within the network. We seek to introduce a framework that allows for a distributed and scalable object classification paradigm among a group of robots.

This paper is organized as follows. After discussing related work in Section II, we introduce the notion of dictionary learning using RGB-D covariance descriptors in Section III along with the proposed framework. Experimental results on a publicly available RGB-D database are provided in Section IV followed by concluding remarks regarding future work in Section V.

II. Related Work

This paper explores the use of covariance descriptors and dictionary learning for the purpose of object classification using RGB-D point cloud data. A large amount of research has been performed on feature descriptor development and machine learning classifiers. Since we present a new feature descriptor and classifier combination, we briefly review past and current approaches to the extraction and classification of point cloud data.

Early work on feature descriptors for point cloud data include spin images introduced by Johnson [9]. Covariance descriptors were introduced to the image processing community by Tuzel *et al.* [10]. Porikli *et al.* [11] have successfully used covariance descriptors for tracking objects in videos. Fehr [12] has shown that not all features that make up a covariance descriptor are useful, i.e. some features

The authors are with the Department of Computer Science and Engineering, University of Minnesota, Minneapolis, USA. Emails: {beksi, npapas}@cs.umn.edu.

may be more discriminative while other features provide no additional discriminative power.

Lai *et al.* [13] introduced several techniques for RGB-D based object detection and recognition using linear support vector machine, Gaussian kernel support vector machine, and random forest classifiers. Additional approaches to object classification using sparse distance learning were developed by the same authors [14]. Kernel descriptors for object recognition were introduced by Bo *et al.* [15]. Bo *et al.* [16] also developed hierarchical kernel descriptors that recursively form image-level features to generate features from pixel attributes.

Sparse coding has been successfully applied to problems in image processing [17] and more recently to feature learning in computer vision. Sparse representations were shown to outperform conventional representations, i.e. raw image patches, in the construction of image-level features by Lee *et al.* [18]. A spatial pyramid sparse coding model that learns sparse representations over SIFT features and achieves a high level of performance on several standard object recognition tasks was developed by Yang *et al.* [19].

A dictionary learning framework for face recognition that can handle errors due to occlusion and corruption is provided by Wright *et al.* [20]. Ramirez *et al.* [21] describe a clustering framework using learned dictionaries suitable for classifying large datasets. Dictionary learning is employed by Blum *et al.* [22] in order to find relevant features for object classification. Bo *et al.* [23] apply previous ideas of hierarchical models with dictionary learning in their work.

III. Dictionary Learning using RGB-D Covariance Descriptors

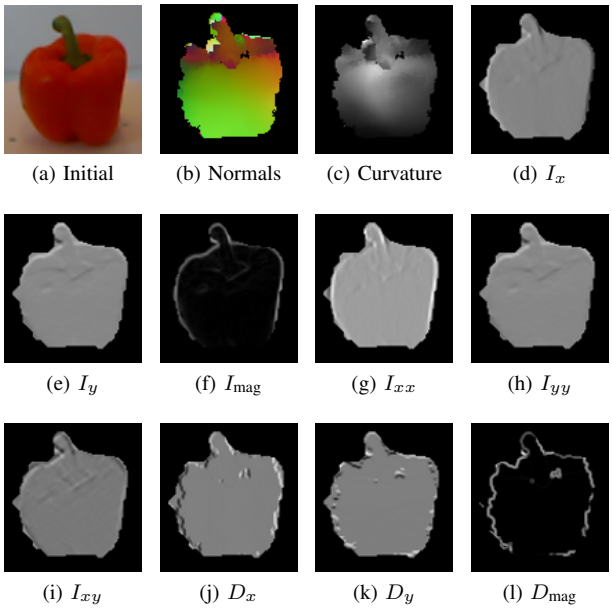
The problem of efficiently extracting discriminative features from an object in a large point cloud dataset and accurately classifying that object is nontrivial. Furthermore, object classification needs to be performed quickly to be of practical use. To facilitate such a process, we compute covariance descriptors that encapsulate features (position, color, normals, etc.) over the entire point cloud of an object. This results in a symmetric positive definite matrix that embodies the object. These covariance descriptors are then used to create dictionaries representing the object. Finally, a set of dictionaries for a range of objects can be used to classify new point cloud data.

A. Covariance Descriptor

Each object is characterized by a single positive definite matrix which we call a covariance descriptor. More formally, let $f_i \in \mathbb{R}^p$, for $i = 1, 2, \ldots, n$, be a feature vector consisting of the n points of an object. Then, the covariance descriptor of the object is defined as

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

where μ_f is the mean feature vector and $C \in \mathcal{S}^p_{++}$ is the space of $p \times p$ Symmetric Positive Definite (SPD) matrices. This representation allows for the compact description of the object of interest. The fact that the covariance is a positive definite matrix allows us to reduce the size of the stored values to the number of upper or lower triangular entries in the matrix. In addition, adding a new feature limits the growth of the matrix to a new row and column.



(a) Initial (b) Normals (c) Curvature (d) I_x

(e) I_y (f) I_{mag} (g) I_{xx} (h) I_{yy}

(i) I_{xy} (j) D_x (k) D_y (l) D_{mag}

Fig. 1. These features, (b)-(l), make up the RGB-D covariance descriptors as described in Section III-B.

Since covariance descriptors occupy a space that spans a Riemannian manifold, we approximate their distances using the log-Euclidean metric introduced by Arsigny *et al.* [24],

$$d(C_1, C_2) = \|\log(C_1) - \log(C_2)\|_F, \tag{2}$$

where C_1 and C_2 are two positive definite matrices, $\log(\cdot)$ is the matrix logarithm, and $\|\cdot\|_F$ is the Froebinius norm.

B. Features

In this work, covariance descriptors are derived from two feature vectors composed of shape and visual invariants respectively (Fig. 1):

$$\begin{aligned} f_{pnkK} &= [x, y, z, n_x, n_y, n_z, k_1, k_2, K] \\ f_{pcimd} &= [x, y, z, r, g, b, I_x, I_y, I_{xx}, I_{yy}, I_{xy}, I_{\text{mag}}, \\ &\quad D_x, D_y, D_{\text{mag}}]. \end{aligned}$$

The first feature vector, f_{pnkK}, consists of nine elements that describe the shape of the object. We estimate the coordinates of the surface normal, n_x, n_y, n_z, by performing an eigendecomposition of the covariance matrix created from the nearest neighbors of a query point for a fixed radius. After the normals are computed at each point, the principal curvatures k_1 and k_2, can be estimated. Curvatures are estimated by projecting the normals in the neighborhood of a query point onto the plane tangent to the point. The parameters of the ellipse fitting the projected points are computed yielding the values of the main curvature axes.

The Gaussian curvature is computed as the product of the principal curvatures, $K = k_1 \cdot k_2$.

The second feature vector, f_{pcimd}, is constructed using fifteen elements that provide visual properties relevant to the object. Color information is provided by the r, g, b color channel values. I_x and I_y correspond to an approximation of the gradient by the Sobel operator applied to the grayscale image along x and y respectively. Applying the operator a second time on the image patch results in I_{xx}, I_{yy}, I_{xy}. The magnitude of the gradient of the image patch is defined as $I_{\text{mag}} = \sqrt{I_x^2 + I_y^2}$. For the depth image of each object, the same operations are performed to produce D_x, D_y and $D_{\text{mag}} = \sqrt{D_x^2 + D_y^2}$.

The position of each point in an object, given by its x, y, z Cartesian coordinate, is common to both feature vectors. The features that make up these vectors were chosen based on empirical observations over a range of datasets along with the findings in [12] regarding feature selection. The total number of features that represent the object in a covariance descriptor is given by $\frac{(p+1)p}{2}$ (p is the number of features) due to the symmetry of the matrix.

C. Dictionary Learning

Sparse modeling makes use of a linear combination of data vectors from a dictionary. The dictionary is a collection of atoms that represent a sparse approximation of a covariance descriptor. Concretely, a dictionary is a matrix D of size $n \times k$ represented by a collection of k atoms where an atom is a column vector of length n. In a sparse representation, we approximate a covariance descriptor $\hat{C}$ as a linear combination of the atoms in a dictionary

$$\hat{C} = D\alpha, \tag{3}$$

where α is a sparse coefficient vector. The goal of dictionary learning is to construct a dictionary such that the approximations of many covariance descriptors, i.e. the training set, are as good as possible given a spareness criterion on the coefficients.

During the training phase, we learn an $n \times k$ dictionary D of covariance descriptors C_i for each object subject to the constraint

$$\min_{D,\alpha} \sum_{i=1}^{m} ||C_i - D\alpha_i||^2 + \lambda||\alpha_i||_1, \tag{4}$$

where m is the total number of covariance descriptors and λ is a regularization constant. Dictionaries are computed for objects corresponding to the feature vectors described in Section III-B. These dictionaries, based on shape and visual features, have respective sizes of $9 \times k$ and $15 \times k$.

At prediction time, we classify an object per the minimum error between the input covariance descriptor C and the approximate representation of the descriptor $\hat{C}$ using the loss function

$$\ell(C, D) = \min_{\alpha \in \mathbb{R}^k} ||C - \hat{C}||^2 + \lambda||\alpha||_1. \tag{5}$$

D. Classification Framework

In robotics, it is important that an object classification system be able to determine if an object has been seen before or if it is an occurrence of an entirely new object. The classification framework is based on being able to detect both the category and the instance of a given object. Category classification is defined as determining if a previously unseen object belongs to the same category of objects that have previously been seen (e.g. Red Delicious, Granny Smith, and Honeycrisp are instances of apples that belong to the category 'apple'). Instance classification is defined as determining if an object is physically the same object that has previously been seen.

Instance classification is performed by computing covariance descriptors using the single feature vector f_{pcimd}. Given a set of m objects, we train a dictionary set $D_I = \{D_1, \ldots, D_m\}$. The elements $D_1, \ldots, D_m$ are the dictionaries for the m objects computed using (4). Classification by category is more challenging than instance classification. We note the findings of [13] regarding the combination of shape and visual features for higher overall category classification performance and that some combinations of features may be redundant [12]. Category classification is done by computing covariance descriptors based on both the f_{pnkK} and f_{pcimd} feature vectors. Given n categories of objects, we train two dictionary sets: $D_{C_1} = \{D_1, \ldots, D_n\}$ and $D_{C_2} = \{D_1, \ldots, D_n\}$. The elements of D_{C_1} consist of dictionaries built using the feature vector f_{pnkK} while the elements of D_{C_2} consist of dictionaries built using the feature vector f_{pcimd}. The elements of D_{C_1} and D_{C_2} are computed using (4).

For each object point cloud, we classify both the category and instance of the object. Two covariance descriptors using feature vectors f_{pnkK} and f_{pcimd} are computed along with their sparse representations using (3). To match the category, we iterate over D_{C_1} and D_{C_2} computing the error for each dictionary using (5). We then predict the category based on the dictionary that produces the smallest error between the sets. Similarly, to match an instance of an object we iterate over D_I and compute the error for each dictionary using (5). The dictionary with the minimum error in D_I is used to predict the instance of the object. The minimum error among all dictionaries can be computed in parallel making the prediction procedure computationally fast. A general overview of the framework is depicted in Fig. 2.

IV. Experimental Results

We evaluate the performance of dictionary learning using covariance descriptors on the RGB-D object database [13]. The database consists of 300 common household objects, divided into 51 categories, recorded using an RGB-D sensor, shown in Fig. 3 and Fig. 4. Following the procedure outlined in [13], we conduct three experiments on a subsample of the database. This subsample is created by taking every fifth frame giving approximately 45,000 frames to perform classification.

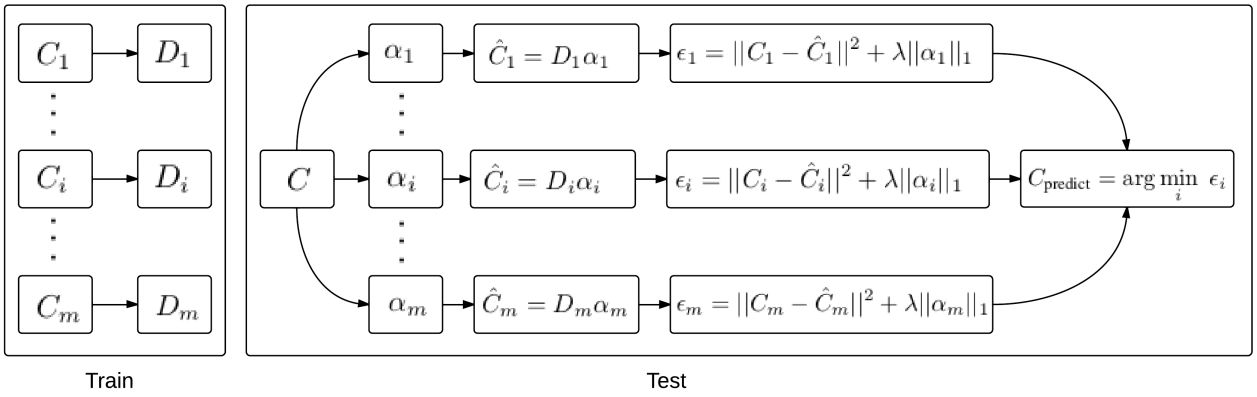


Fig. 2. The classification framework for dictionary learning using RGB-D covariance descriptors. In the training phase, we learn a dictionary D_i for each covariance descriptor set C_i. At testing time, given a set of covariance descriptors C, a prediction is made with respect to the minimum error ϵ among all the dictionaries.

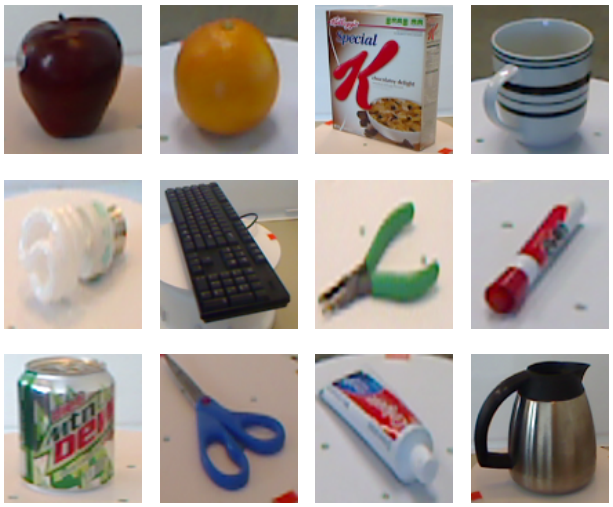


Fig. 3. A subset of the items contained in the database introduced by Lai et al. [13]. Each item is representative of one of the fifty-one categories within the database.

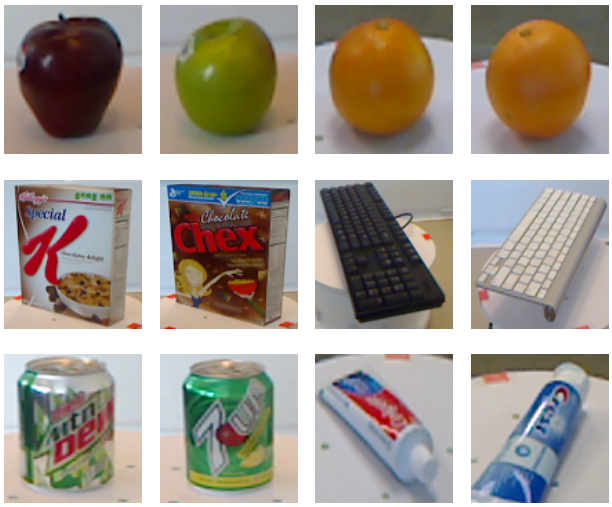


Fig. 4. Columns 1 and 3 represent an instance of an item from a category in the RGB-D database [13]. Columns 2 and 4 represent a different instance of the item within the same category.

The experiments consist of category and instance classification as described in Section V of [13]. For category classification, we randomly leave one object out from each category for testing and train on all views of the remaining objects. For instance classification, two scenarios are considered:

- Alternating contiguous frames: Divide each video into 3 contiguous sequences of equal length. There are 3 heights (videos) for each object which gives 9 video sequences for each instance. We randomly select 7 of these for training and test on the remaining 2.
- Leave-sequence-out: Train on the video sequences of each object where the sensor is mounted 30° and 60° above the horizon and evaluate on the 45° video sequence.

For the cases in which the frames are randomly chosen, the experiments are run 10 times and the average and standard deviation are reported. The dictionary learning routines are provided by the SPAMS optimization toolbox [25]. Table I compares the results of the experiments to other methods.

A. Instance Classification

We perform instance classification by training a single set of dictionaries, $D_I = \{D_1, D_2, \ldots, D_{300}\}$. Each dictionary in D_I is of size 15×64 (64 atoms) and represents one object in the RGB-D database. The dictionaries are computed using the feature vector emphasizing visual features, f_{pcimd}. During testing, the covariance descriptors are computed on

TABLE I

CLASSIFICATION ACCURACY (INSTANCE AND CATEGORY) AND DIMENSIONALITY. (A) LEAVE-SEQUENCE-OUT, (B) ALTERNATING CONTIGUOUS FRAMES. COMPARISON ACCURACIES ARE AVERAGED OVER 10 TRIALS. THE LAST ROW REPORTS THE RESULTS OF THE EXPERIMENTS USING DICTIONARY LEARNING (DL).

Method	Instance		Category	Dim
	(a)	(b)		
Linear SVM [13]	73.9	90.2 ± 0.6	81.9 ± 2.8	4203
Nonlinear SVM [13]	74.8	90.6 ± 0.6	83.8 ± 3.5	4203
Random Forest [13]	73.1	90.5 ± 0.4	79.6 ± 4.0	4203
IDL [14]	-	91.3 ± 0.3	85.4 ± 3.2	4203
HKDES [15]	82.4	-	84.1 ± 2.2	7000
Kernel Desc. [16]	84.5	-	86.2 ± 2.1	39000
CKM Desc. [22]	90.4	92.1 ± 0.4	86.4 ± 2.3	19200
Upgraded HMP [23]	92.8	-	87.5 ± 2.9	188300
Cov Desc. SVM [12]	90.7	94.4 ± 2.0	80.4 ± 1.9	253
Cov Desc. DL	93.7	96.9 ± 0.5	85.7 ± 3.5	165

the input frames of the object along with the minimum errors among all dictionaries in D_I. We then use a majority vote aggregation strategy to classify the object based on the number of votes from each dictionary.

The results in Table I show that this method of instance classification outperforms the results of existing methods on the RGB-D database. Moreover, the covariance descriptor computed per frame exists in $\frac{(15+1)15}{2} = 120$ dimensional space. These results also show that visual features alone provide high instance classification performance and support the findings in [12] and [13].

B. Category Classification

We perform category classification by training two sets of dictionaries: $D_{C_1} = \{D_1, D_2, \ldots, D_{51}\}$ and $D_{C_2} = \{D_1, D_2, \ldots, D_{51}\}$. The dictionaries of D_{C_1} are of size 9×256 (256 atoms) and utilize the feature vector that consists of shape features, f_{pnkK}. The set D_{C_2} contains dictionaries of size 15×256 (256 atoms) represented by the feature vector f_{pcimd}. Each dictionary element contains all instances of an object that make up the object's category minus the instances of the new object. For each test frame, we compute two covariance descriptors using feature vectors f_{pnkK} and f_{pcimd}. The covariance descriptors are then matched against their respective dictionary sets. The votes for the minimum error within each set are aggregated and we classify the object based on the majority vote.

This method of category classification performs fairly well in comparison to other methods (Table I). In addition, the dimensionality is low, for each input frame the object can be represented in $\frac{(9+1)9}{2} + \frac{(15+1)15}{2} = 165$ dimensional space. The results show that shape features are useful in category classification which concur with [12] and [13]. Fig. 5 shows a plot of the confusion matrix generated over all trials of the category classification experiment.

C. Subsampled Category Classification

In this experiment we measure the performance of the classification framework as the number of categories increase.

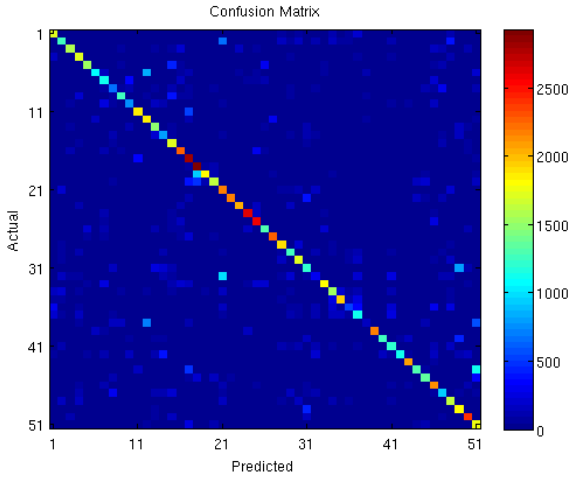


Fig. 5. The category classification confusion matrix generated over 10 trials as outlined in [13].

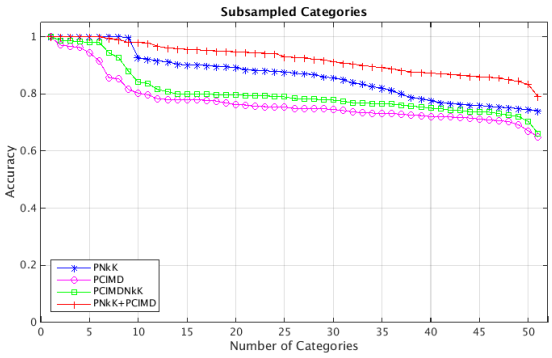


Fig. 6. Subsampled category classification accuracy results using a set of dictionaries consisting of only shape features (PNkK), only visual features (PCIMD), both shape and visual features (PCIMDNkK), and two sets of dictionaries using shape and visual features independently (PNkK+PCIMD).

We also compare the performance against three different classification setups. The first setup uses a single set of dictionaries consisting only of shape features (f_{pnkK}) to perform category classification. The second setup uses one set of dictionaries with only visual features (f_{pcimd}). The third setup consists of a set of dictionaries that combine both shape and visual features using the feature vector $f_{pcimdnkK}$. All dictionaries are composed of 256 atoms. A comparison of the results is shown in Fig. 6.

The results of this experiment highlight several important findings. The first finding is that the shape of an object provides useful information in category classification which agrees with [12] and [13]. Second, in a dictionary learning environment, combining all features into a single dictionary does not increase the classification performance. Third, the use of smaller dictionaries with distinct features employing a majority voting scheme leads to more stable and overall better performance. It also indicates that within the RGB-D database, certain objects are classified more accurately by visual features than shape, and vice versa.

The third finding is especially important for object clas-

sification in robotics. Many smaller dictionaries are more amenable to parallel computations compared to having fewer larger dictionaries. These dictionaries can be readily stored, accessed, and transferred over a robotic network. The adaptability of a dictionary learning classifier in this framework is apparent compared to other classifiers, such as a support vector machine (SVM), where a single model would need to be centrally shared among robots and updated atomically.

V. Conclusion and Future Work

This paper has presented a framework that incorporates dictionary learning with RGB-D covariance descriptors on point cloud data for performing object classification. The purpose of this framework is to introduce a compact and flexible feature descriptor combined with a highly accurate classifier that has the potential of being deployed in either an ad hoc or cloud robotic network. In terms of size, we only need to store 45 shape parameters and 120 visual parameters per point cloud. The classifier exhibits versatility in that updating the framework amounts to adding a new dictionary to the existing set of dictionaries.

There are a number of open questions that remain from this work. How does the framework handle the detection of a new object? A possible solution is to set an error threshold such that no matching dictionary indicates a new object has been detected. Training a set of dictionaries grows significantly in computation time as the number of objects and/or atoms increase. To mitigate this problem, training can be done offline in an environment with suitable computational resources. When creating dictionaries, it's unclear how to choose the optimal number of atoms. Too few or too many atoms per dictionary can adversely affect classification performance.

Our interests lie in developing scalable object classification capabilities among heterogeneous groups of robots. Going forward, future work includes the addition of this classification framework into a general purpose object recognition engine. We envision robots in a factory or a household performing classification tasks utilizing our dictionary learning framework.

Acknowledgments

This work was carried out in part using computing resources at the University of Minnesota Supercomputing Institute. This material is based in part upon work supported by the National Science Foundation through grants #IIP-0934327, #IIS-1017344, #CNS-1061489, #CNS-1138020, #IIP-1332133, #IIS-1427014, and #IIP-1432957.

References

[1] Kinect. [Online]. Available: http://www.xbox.com/en-us/kinect

[2] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2010, pp. 356–369.

[3] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (fpfh) for 3d registration," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2009, pp. 3212–3217.

[4] F. Tombari, S. Salti, and L. Di Stefano, "A combined texture-shape descriptor for enhanced 3d feature matching," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2011, pp. 809–812.

[5] R. B. Rusu, G. Bradski, R. Thibaux, and J. Hsu, "Fast 3d recognition and pose using the viewpoint feature histogram," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010, pp. 2155–2162.

[6] 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 IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 1793–1798.

[7] D. Fehr, W. J. Beksi, D. Zermas, and N. Papanikolopoulos, "Rgb-d object classification using covariance descriptors," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 5467–5472.

[8] B. A. Olshausen and D. J. Field, "Sparse coding with an overcomplete basis set: A strategy employed by v1?" *Vision research*, vol. 37, no. 23, pp. 3311–3325, 1997.

[9] A. E. Johnson, "Spin-images: A representation for 3D surface matching," Ph.D. dissertation, Carnegie Mellon University, 1997.

[10] O. Tuzel, F. Porikli, and P. Meer, "Region covariance: A fast descriptor for detection and classification," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2006, pp. 589–600.

[11] F. Porikli, O. Tuzel, and P. Meer, "Covariance tracking using model update based on lie algebra," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 2006, pp. 728–735.

[12] D. Fehr, "Covariance based point cloud descriptors for object detection and classification," Ph.D. dissertation, University of Minnesota, 2013.

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

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

[15] L. Bo, K. Lai, X. Ren, and D. Fox, "Object recognition with hierarchical kernel descriptors," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 1729–1736.

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

[17] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, 2006.

[18] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in *Advances in neural information processing systems*, 2006, pp. 801–808.

[19] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 1794–1801.

[20] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[21] I. Ramirez, P. Sprechmann, and G. Sapiro, "Classification and clustering via dictionary learning with structured incoherence and shared features," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2010, pp. 3501–3508.

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

[23] L. Bo, X. Ren, and D. Fox, "Unsupervised feature learning for rgb-d based object recognition," in *Experimental Robotics*. Springer, 2013, pp. 387–402.

[24] 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.

[25] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *The Journal of Machine Learning Research*, vol. 11, pp. 19–60, 2010.