3D Object Categorization and Recognition based on Deep Belief Networks and Point Clouds

Fatima Zahra Ouadiay, Nabila Zrira, El Houssine Bouyakhf and M. Majid Himmi
LIMIARF, Faculty of Sciences, Mohammed V University, Rabat, Morocco

Keywords: Real 3D Object Recognition, Categorization, Deep Belief Network, PCL, 3D SIFT, SHOT, CSHOT.

Abstract: 3D object recognition and categorization are an important problem in computer vision field. Indeed, this is an area that allows many applications in diverse real problems as robotics, aerospace, automotive industry and food industry. Our contribution focuses on real 3D object recognition and categorization using the Deep Belief Networks method (DBN). We extract descriptors from cloud keypoints, then we train the resulting vectors with DBN. We evaluate the performance of this contribution on two datasets, Washington RGB-D object dataset and our own real 3D object dataset. The second one is built from real objects, following the same acquisition conditions than those used for Washington dataset acquisition. By this proposed approach, a DBN could be designed to treat the high-level features for real 3D object recognition and categorization. The experiment results on standard dataset show that our method outperforms the state-of-the-art used in the 3D object recognition and categorization.

1 INTRODUCTION

We live in a 3D world. We can recognize and name an object without any difficulty. The specific knowledge stored in a brain, allow us to compare the information of the object presented in the scene with those stored to detect and recognize the object without any ambiguity.

In the field of robotics, to apply this principle of vision and develop the object manipulation aspect in the Human-Machine Interaction (HMI), researchers have developed some algorithms that provide the steps of human vision procedure. The most popular methods are: the extraction of information from scenes by using some detectors/descriptors algorithm such as (SIFT, SURF, ORB, ect) (Lowe, 1999), (Bay et al., 2006).

Several works have been developed in this area, especially with introduction of 3D Library as PCL in 2012 (Aldoma et al., 2012). The most of the work must use the large database of several perspective 3D objects.

Recently, the robotic researchers seek to integrate machine learning methods for robotic tasks. The methods that are based on deep learning (Bengio, 2009) demonstrated the performance of state-of-the-art in a wide variety of tasks, including visual recognition (Le, 2013), natural language processing (Collobert et al., 2011), and the speech recognition (Hinton et al., 2012). These techniques are particularly powerful because they can learn useful features directly from unlabeled and labeled data, eliminating the need for hand-engineering. However, most of the works in deep learning were in pedestrian detection (Sermanet et al., 2013). The offered approaches on the object recognition still require a dataset as a basis for references or for tests. Some Researchers consrtuct their object data to assess their methods as Liang (Liang et al., 2014) and Yu (Yu et al., 2013). Some others used one of the popular datasets that are available for public by other researchers as Scwarz (Schwarz et al., 2015) and Alexandre (Alexandre, 2016a). NORB (Nair and Hinton, 2009a). Washington RGB-D dataset (Lai et al., 2011a) is an another data which is the most used, because it has a very wide selection (categories and instances) of real objects of interior environment.

Our goal in this paper is to propose an object categorization and recognition approach using Deep belief network (DBN) in the robotic gripper problem in a way that could be generalized to similar problems of detection and recognition. The main objective is to recognize the target object in the scene based on the specific characteristics that identify the class to which it belongs. We test our approach on different types of dataset, firstly on reference dataset (Washing-

Ouadiay, F., Zrira, N., Bouyakhf, E. and Himmi, M.
3D Object Categorization and Recognition based on Deep Belief Networks and Point Clouds.
DOI: 10.5220/0005979503110318
In *Proceedings of the 13th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2016) - Volume 2*, pages 311-318
ISBN: 978-989-758-198-4

ton RGB-D object dataset) and secondly on our own real 3D object dataset. Our 3D acquisition system is presented in section 4.2 to validate generally and in real manner the proposed method. These experiments can show that our method improves the performance of categorization and recognition to manipulate any object.

The rest of the paper is organized as follows:

We describe related work in Section II. Section III, presents our contribution, and some additional details for each step of contribution. In Section IV, we illustrate the datasets. Feature extraction is introduced in section V. Then we describe our learning algorithm feature DBN in Section VI. We present the experiments and results in the Section VII. We close with several interesting directions for future work and conclude this work in Section VIII.

2 RELATED WORK

Recently, researchers have been interested in 3D object recognition due to the development of RGB-D cameras that provide a high quality synchronized depth and color data. In (Bo et al., 2011), the authors develop a set of kernel features over depth maps that model 3D shape, size, and depth edges. The main match kernel framework defines pixel attributes, designs match kernels in order to measure the similarities of image patches, then determines low dimensional match kernels. In (Savarese and Fei-Fei, 2007), authors suggest a compact model of 3D object category based on appearance and 3D geometric shape. Each object is considered as a linked set of parts that are composed of many local invariant features. The approach can classify, localize and infer the scale as well as the pose estimation of objects in the image. In (Toldo et al., 2009), authors introduce Bag of Words (BoW) approach for 3D object categorization. Spectral clustering is used to select seed-regions, followed by hierarchical clustering at each level for region descriptors in order to obtain BoW histograms for each mesh. Finally, Support Vector Machine (SVM) is learnt to classify different BoW histograms for 3D objects. In (Lai et al., 2011b), the authors describe the dataset collection steps and propose methods to recognize and detect RGB-D objects. They use spin image descriptor to extract shape features that are used for computing efficient match kernel (EKM). They use also SIFT descriptor to extract visual features. Finally, linear support vector (LiSVM), gaussian kernel support vector machine (kSVM) and random forest (RF) are learnt to classify both color and depth informations. In (Nair and Hinton, 2009b), a new 3D object recognition approach is proposed and tested on NORB database. The dimensionality for each stereo-pair image is reduced by using a foveal image. The final representation is equal to 8976 dimensional vectors that are learnt with a top-level model for Deep Belief Nets (DBN). This model is a third-order Boltzmann machine which is trained using a hybrid algorithm that combines both generative and discriminative gradients. The first convolutional-recursive deep learning model is introduced in (Socher et al., 2012) for 3D object recognition. The authors compute a single CNN layer to extract low level features from both RGB and depth images. These representations are given as input to a set of RNNs with random weights. The concatenation of all the resulting vectors forms the final feature vector for a softmax classifier. In (Alexandre, 2016b), author propose a new approach for RGB-D object classification. He uses four independent Convolutional Neural Networks (CNNs), one for each channel, then train these CNNs in a sequence. The authors of (Schwarz et al., 2015), provide a meaningful feature set that results from the pre-trained stage of Convolutional Neural Network (CNN). Then, they incorporate depth information which is not trained with CNN. Depth and RGB images are processed independently by CNN and the resulting features are used to determine category, instance and pose of the object.

Our work focuses on 3D object representation as well as recognition and categorization using 3D PCL descriptors and Deep Belief Networks (DBNs). We extract 3D keypoints with 3D SIFT detector which are described using SHOT and CSHOT descriptors. Resulting vectors are learnt using Deep Belief Networks (DBNs) classifier.

3 OVERVIEW OF OUR CONTRIBUTION

In this article, we propose a new system that can classify and recognize objects. The most works, that are done on object recognition based on machine learning methods, test their methods using testing and training sets from the same dataset. We also evaluate the experiment on a real object dataset in order to prove the approach validation in any indoor environment. To the best of our knowledge, this study is the first to do.

To improve the capacity of recognition and categorization methods, we propose to focus feature learning using Deep Belief Network. We train only the most interesting points of the point cloud that represents the sought object. For that, we introduce a preprocessing step to extract keypoints by the 3D SIFT

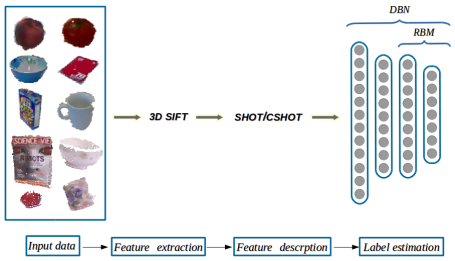

Figure 1: An overview of our model: from the input data, we describe keypoins extracted by 3D SIFT with SHOT and CSHOT, after this step, the objects are ready for DBN process.

detector which gives the best results on several papers (Alexandre, 2012). Then, the approach can identify the significant features using the best descriptors in terms of time computing and recognition rates SHOT and SHOTcolor (CSHOT).

Figure 1 gives an overview of our approach. To use a DBN, the data of object must be carefully prepared. The input dataset must be segmented and filtered to reduce parasitic elements that appear during real time object acquisition.

The Washignton RGB-D dataset is already segmented, however we ameliorate the quality of our acquired data with the Meshlab software that was developed in the Visual Computing laboratory. It implements a wide range of algorithms and filters that improve the reconstruction of 3D models. After this step, the objects are ready for preprocessing. The features contained in the point clouds are picked by order of importance using the 3D SIFT detector. The keypoints chosen by SIFT detector are identified by the SHOT and CSHOT descriptors to evaluate the influence of color information. The descriptors are then adjusted to the input distribution of the DBN network to predict the object class.

In summary, our major contributions are as follows:

- We introduce a novel pipeline for RGB-D object recognition and categorization that combines point clouds processing and DBN;
- We pick and analyze the important point before learning the DBN with 3D SIFT and SHOT/CSHOT descriptors;
- We demonstrate the validation of our approach on two datasets:
  1. Washington dataset is used for recognition test: each class contains different views of the same object;
  2. Washington dataset is used for categorization test: each category contains three instances of object.
- Finally, we demonstrate that our method improves the recognition rate compared to other methods.

4 DATASET

Two sets of 3D indoor object data are used to evaluate the proposed approach for object recognition and categorization. The first one is the large famous Washington RGB-D dataset represented in figure 2, which is used for the training data of both experiments. The second one is used for the testing in the recognition model with real objects. It is our own dataset represented in figure 4 that is acquired in the same conditions as Washington dataset using our reconstruction system equipped with the RGBDemo software.

4.1 Washignton RGB-D Dataset

We use a subset of the RGB-D washington dataset from (Lai et al., 2011a), this data contains 300 objects that are organized into 51 categories.

For the recognition test, we use 10 classes of objects, each class contains different clouds of the same object captured from different points of view.

For the categorization test, we use the same classes of objects, each category contains three different instances (example: the apple category contains green, red and yellow apples).

Figure 2: The ten object classes used in our experiments from Washington RGB-D dataset (apple(1), bowl(2), calculator(3), cellphone(4), coffee-mug(5), tomato(6), food-bag(7), food-box(8), marker(9), notebook(10)).

4.2 Our Real RGB-D Dataset

The 3D acquisition system of 3D object models aims to gather and represent the information associated with a real-world object using multiple views as captured by Kinect.

The main purpose is to build a rotating support while keeping the Kinect camera fixed.

The system consists of Kinect camera, software and hardware parts:

The hardware is composed of rotating support that

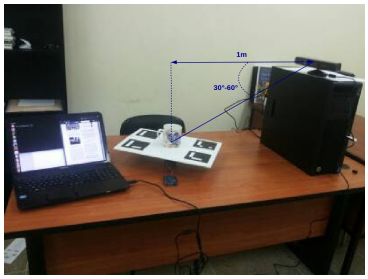

Figure 3: The 3D acquisition system contains: Kinect camera, RGBDemo software and rotating support commanded by arduino Kit.

consists of four fiducial markers aligned at fixed positions to form a rectangular shape. The board is actuated by a precise motor, so that we know the pose of the object at each Kinect frame. We use a stepper motor to achieve very precise positioning and speed control. For the precision motion, the stepper motor is maintained using Arduino Kit and Adafruit Motor Shield.

The software is RGBDemo that provides a simple toolkit to start fusion with Kinect data and develop standalone computer vision programs. The project consists of a library called nestk, which is designed to easily integrate into existing cmake-based software and provides quick access to the Kinect features. It includes OpenCV for image processing, QT for the graphical parts, libfreenect for Kinect, and PCL library. The main idea of the demonstration is to build a 3D model for real-world objects using open source Aruco library (BSD licenced), that is able to generate and recognize square markers, issue the ID and the corner cordinates of each detected marker.

To generate the viewpoints in the same condition us whashington dataset, in figure 3 we fix the kinect's position during the movement of the support, and it's placed about one meter from the turntable (the minimum distance required for the RGB-D camera to return reliable depth readings) in order to ensure a constant illumination and avoid the risk of having desynchronized depth and color images. For each class, we take maximum data to have an extensive object views in 360°, with the camera mounted at different heights relative to the rotating support at approximately between [30°,60°] to validate the approach independently regardless of view angles.

5 FEATURE EXTRACTION

5.1 3D Sift Detector

Scale-invariant feature transform (SIFT) is an algorithm deployed in the field of computer vision to detect and describe regions in an image and identify similar elements between varying images called matching. The algorithm consists of the detected feature points of an image used to characterize every point that needs to be recognized by comparing its characteristics with those of the points contained in other images. The general idea of SIFT is to find the keypoints that are invariant to several transformations/changes: rotation, scale, illumination and viewing angle. The 3D SIFT detector (Lowe, 1999) use the Difference-of-Gaussian (DoG) function to extract the extrema points in both spatial and scale dimensions.

Figure 4: The object classes (tomato,lemon,coffee-mug,bowl,notebook) from our acquired RGB-D dataset: 2D presentation (in down), point cloud (in top).

5.2 SHOT/CSHOT Descriptors

The SHOT (Signature of Histograms of Orientations), proposed by Tombari and al. in (Tombari et al., 2010), (Tombari et al., 2011) is a descriptor based on the histograms of normal. It is established from the intersection between signatures and histograms, so as to achieve a better balance between the descriptive character and the robustness. In addition, it presents the descriptive power of the 3D shape of the surface that was repeatable and robust to noise, translations, and rotations. It presents an enormous gain in computing time. The description of the geometrical information about the point positions contained in a support is made by a set of local 3D histograms defined on a 3D spherical grid that partitions the space according to the radial axes, azimuth, and elevation. For each sector of the grid the values of the cosine of angles between the normal reference and all these neighbors are accumulated to form the normal histogram with 32 bins. The estimation of the normal is made by calculating a new covariance matrix as linear combination of the distances of the points belonging to a spherical support of the keypoints. The eigenvectors of this matrix from orthogonal directions are repeatable and robust to noise. It is possible to improve the discriminating power of the descriptor by introducing geometrical information concerning the location of points inside the support, in order to obtain a signature. It makes by calculating a first set of local histograms on 3D volumes defined by a 3D grid overlaid

on the support and then grouping all local histograms to form the resulting descriptor.

More recently, SHOTCOLOR (CSHOT) version combines SHOT information on the shape, texture and colors. This descriptor is a combination of a normal histogram and a color one. The color histogram is formed by RGB absolute values between the reference point and their neighboring ones.

6 DEEP BELIEF NETWORKS

6.1 Restricted Boltzmann Machines

Restricted Boltzmann machines (RBMs) (Smolensky, 1986) are a particular type of energy-based model with hidden variables. They are restricted in the sense that no variable-variable or hidden-hidden connections exist. As shown in Figure 5, RBMs are undirected graphical models that are composed of two layers:

1. The first layer: it contains visible units (x) that correspond to the components of an observation (i.e. SHOT/CSHOT descriptors in this case of study);
2. The second layer: it contains hidden units (h) that model dependencies between the components of observations.

The energy function of an RBM is defined as:

$$E(x,h) = -b^{'}x - c^{'}h - h^{'}Wx \qquad (1)$$

where:

- W: represents the symmetric interaction term between visible units (x) and hidden units (h);
- a and b : are vectors that store the visible (input) and hidden biases (respectively).

Then, we introduce the notation (inspired from physics) of free energy in order to marginalize energies in the log-domain. The following free energy formula can be written as follows:

$$F(x) = -b^{'}x - \sum_{i} log \sum_{h_i} e^{h_i(c_i + W_i x)} \qquad (2)$$

RBMs have received a lot of attention recently after being proposed as building blocks of multi-layer learning architectures called deep belief networks. The idea is that the hidden neurons extract relevant features from the observations. These features can serve as input to another RBM. By stacking RBMs in this way, one can learn features from features in the hope of arriving at a high level representation.

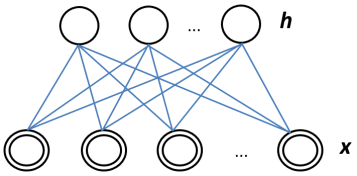

Figure 5: RBM models. The joints between hidden units and also between visible units are disconnected.

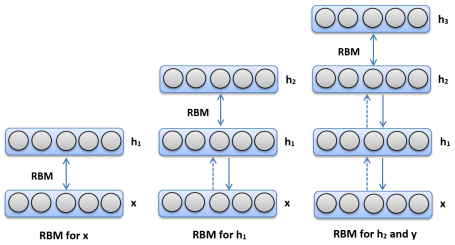

Figure 6: DBN framework: three hidden layers h1, h2, h3 and one visible layer x.

6.2 Deep Belief Networks

Deep Belief Networks (DBNs) are probabilistic generative models with many layers of stochastic and hidden variables (Deng and Yu, 2014). In (Hinton et al., 2006), the authors introduce the motivation for using a deep network versus a single hidden layer (i.e. a DBN vs an RBM). The power of deep networks is achieved by having more hidden layers.

The DBN architecture is composed of the top two layers that are undirected with symmetric connections between them. This block represents a probabilistic model called a Restricted Boltzmann Machine. Whereas the lower layers are directed connections from the layer above. Figure 6 shows typical DBN with one input layer (x) and three hidden layers h1,h2 and h3. In the first stage of DBN training processes, each pair of layers grouped together to reconstruct the input of the layer from the output. The layer-wise reconstruction happens between x and h1, h1 and h2, h2 and h3, respectively, which is implemented by a family of RBMs. After the greedy unsupervised learning of each pair of layers, the features are progressively combined from loose low-level representations into more compact high-level representations. In the second stage, the whole deep network is then refined using a contrastive version of the wake-sleep algorithm via a global gradient-based optimization strategy.

7 RESULTS AND DISCUSSION

In this section, we tested our 3D recognition and categorization approaches on both Washington RGB-D

as well as our real RGB-D datasets. The training and testing point clouds are computed using a Xeon(R) 3.50 GHz CPU 32 Go RAM K2000 Nvidia card on Ubuntu 14.04. DBN aims to allow each RBM model in the sequence to receive a different representation of the data. In other words, after RBM has been learned, the activity values of its hidden units are used as the training data for learning a higher-level RBM.

In this work, we use the SHOT/CSHOT descriptors to extract features from point clouds, which are considered as the input layer x of DBN architecture in figure 6. The input layer has a number N of units, equal to the size of sample data x (352 for Shot and 1344 for ShotColor) . The number of units for hidden layers, currently, are pre-defined according to the experiment. We fixed DBN with two hidden layers h1 and h2. The general DBN characteristics are shown in Table 1.

Table 1: DBN characteristics that are used in our experiment.

Characteristic	Value
Hidden layers	2
Hidden layer units	600
Learn rates	0.3
Learn rate decays	0.9
Epochs	50
Verbose	1
Input layer units	size of descriptor

7.1 3D Object Recognition

In this sub-section, we evaluate the performance of 3D object recognition system on Washington RGB-D data with classes (apple(1), bowl(2), calculator(3), cellphone(4), coffee-mug(5), tomato(6), food-bag(7), food-box(8), marker(9), notebook(10)) and our real RGB-D dataset. We use a DBN with two hidden layers. Then, we train the weights of each layer separately with the fixed number of epochs equal to 50. The approach trains RBMs one after another and uses their resulting training data for training stage in the next RBM.

7.1.1 Washington RGB-D Dataset

Table 2 shows the results of 3D object recognition approach that utilizes the SHOT descriptor for describing 3D keypoints. We remark that the approach can confuse some objects which seem similar (i.e.calculator(3)/cell phone(4), food-box(7)/food-bag(8)). This result is due to the similarity between some object views. Indeed, the very thin side of cell phone could be considered as a thin side of calculator. They are very similar because both contain the keybord. It shows also the results of 3D object recognition approach using CSHOT descriptor. It is very obvious that when we add RGB information to depth data, the recognition accuracy increases. In this case, our approach is more consistent and can not confuse the objects. As shown in Table 4, our approach outperforms all methods that are mentioned in state-of-the-art. The uses of CSHOT descriptor works perfectly with accuracy rate of 99.7% on Washington RGB-D dataset.

7.1.2 Our Real RGB-D Dataset

The aim of our contribution is to exploit 3D object recognition and categorization approach in real-time applications. For this purpose, we constructed our own database from our real objects. We collect four class types (bowl, coffe-mug, tomato and notebook) which are present on Washington RGB-D dataset.

Table 5 shows results on our real-world environment. We remark that the accuracy of our approach using just depth information is better than the one which combines depth and RGB informations (color). This result is evident because in this experiment, we used Washington RGB-D dataset for training stage, while in the testing stage, we used our real objects. So, when objects (test and these in reference class) have the same shapes but color informations are very different. However, the uses of depth only, allow to the system to give a good recognition.

7.2 3D Object Categorization

In this sub-section, we tested 3D categorization approach on Washington RGB-D dataset. Each category, contains three instances of objects. Similarly to above approach, we first test our approach using SHOT descriptor then we add RGB informations.

Table 3 (left) illustrates the results of 3D object categorization using depth information. Our approach can confuse some objects that have the similar shapes (apple(1)/tomato(6), cell-phone(4)/marker(9) and calculator(3)/cell phone(4)), and between (food-box(8)/food-bag(7). When we add RGB information, our approach confuse just some objects who have the similar view angles,(calculator(3) and cell-phone(4)) (see table 3 (right)).

Table 6 demonstrates that our approach outperforms all state-of-the-art methods in both depth and depth with RGB informations.

In general, we can conclude that our approach of 3D object recognition and categorization outperforms all state-of-the-art methods. The RGB informations have an important impact on accuracy results.

Table 2: Performance of 3D object recognition approach using various descriptors SHOT/SHOTCOLOR tested on Washington RGB-D dataset. We use 3D SIFT to extract features (obj/c: number of objects in each test class).

Classes	SHOT						CSHOT					
	obj/c	TP	FN	wrong class	recall	precision	obj/c	TP	FN	wrong class	recall	precision
(1)	207	207	0	–	100%	100%	224	244	0	–	100%	100%
(2)	183	183	0	–	100%	100%	211	211	0	–	100%	100%
(3)	200	188	12	(4)	94%	92%	196	193	0	–	98%	100%
(4)	184	164	20	(3)	89%	99%	155	154	1	(3)	99%	99%
(5)	186	186	0	–	100%	100%	189	189	0	–	100%	99%
(6)	212	212	0	–	100%	100%	211	211	0	–	100%	100%
(7)	256	255	1	(8)	100%	99%	250	250	0	–	100%	100%
(8)	249	249	0	–	100%	100%	250	250	0	–	100%	99%
(9)	269	269	0	–	100%	96%	250	250	0	–	100%	100%
(10)	267	267	0	–	100%	100%	277	277	0	–	100%	100%
Average	–	–	–	–	98.8%	98.6%	–	–	–	–	99.7%	99.7%

Table 3: Performance of 3D object categorization approach using various descriptors SHOT/SHOTCOLOR tested on Washington RGB-D dataset. We use 3D SIFT to extract features (obj/c: number of objects in each test class).

Classes	SHOT						CSHOT					
	obj/c	TP	FN	wrong class	recall	precision	obj/c	TP	FN	wrong class	recall	precision
(1)	207	192	15	(6)	93%	99%	203	203	0	–	100%	100%
(2)	197	197	0	–	100%	100%	188	188	0	–	100%	100%
(3)	204	201	3	(4)	99%	93%	188	188	0	–	97%	100%
(4)	188	168	20	(3)	89%	97%	181	167	14	(3)	95%	92%
(5)	186	186	0	–	100%	100%	191	191	0	–	100%	100%
(6)	197	195	2	(1)	99%	92%	213	213	0	–	100%	100%
(7)	188	179	9	(8)	95%	100%	218	218	0	–	100%	100%
(8)	194	190	4	(7)	98%	99%	184	184	0	–	100%	100%
(9)	213	210	3	(4),(3)	99%	95%	198	194	4	(4),(3)	98%	98%
(10)	201	201	0	–	100%	98%	211	211	0	–	100%	100%
Average	–	–	–	–	97.2%	97.3%	–	–	–	–	99%	99%

Table 4: Comparison 3D object recognition accuracies on the Washington RGB-D Objects dataset.

Approach	Depth	RGB-D
(Lai et al., 2011a)	51.2%	90.6%
(Bo et al., 2011)	54.3%	84.5%
(Schwarz et al., 2015)	–	94.1%
Our	98.6%	99.7%

Table 5: 3D object recognition accuracies on our real RGB-D objects dataset.

Approach	Depth	RGB-D
Real RGB-D dataset	72%	54%

Table 6: Comparison 3D object categorization accuracies on the Washington RGB-D Objects dataset.

Approach	Depth	RGB-D
(Lai et al., 2011a)	64.7%	83.8%
(Bo et al., 2011)	78.8%	86.2%
(Schwarz et al., 2015)	–	89.4%
Our	97.3%	99%

8 CONCLUSIONS AND OUTLOOK

In this paper, we focused on 3D object recognition and categorization using 3D local features which are extracted from PCL descriptors. These features are learned with Deep Belief Netwroks (DBNs) classifier that are based on Restricted Boltzmann machine (RBM).

We tested our approach on both RGD-B Washington dataset as well as our real-world 3D objects. The experimental results are encouraging, especially that our approach is able to recognize and categorize 3D objects under different views.

In future work, we will try to expand our 3D real object data in order to put it available to researchers, then to integrate our algorithm after having improved it in a mobile robot so that, recognize, locate and manipulate objects in real time. We will also compare our approach with the overall characteristics and CNN architecture.

REFERENCES

Aldoma, A., Marton, Z.-C., Tombari, F., Wohlkinger, W., Potthast, C., Zeisl, B., Rusu, R. B., Gedikli, S., and Vincze, M. (2012). Point cloud library. *IEEE Robotics & Automation Magazine*, 1070(9932/12).

Alexandre, L. A. (2012). 3d descriptors for object and category recognition: a comparative evaluation. In *Workshop on Color-Depth Camera Fusion in Robotics at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vilamoura, Portugal*, volume 1, page 7. Citeseer.

Alexandre, L. A. (2016a). 3d object recognition using convolutional neural networks with transfer learning between input channels. In *Intelligent Autonomous Systems 13*, pages 889–898. Springer.

Alexandre, L. A. (2016b). 3d object recognition using convolutional neural networks with transfer learning between input channels. In *Intelligent Autonomous Systems 13*, pages 889–898. Springer.

Bay, H., Tuytelaars, T., and Van Gool, L. (2006). Surf: Speeded up robust features. In *Computer vision–ECCV 2006*, pages 404–417. Springer.

Bengio, Y. (2009). Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127.

Bo, L., Ren, X., and Fox, D. (2011). Depth kernel descriptors for object recognition. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 821–826. IEEE.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.

Deng, L. and Yu, D. (2014). Deep learning: Methods and applications. *Foundations and Trends in Signal Processing*, 7(3–4):197–387.

Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., et al. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine, IEEE*, 29(6):82–97.

Hinton, G. E., Osindero, S., and Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554.

Lai, K., Bo, L., Ren, X., and Fox, D. (2011a). A large-scale hierarchical multi-view rgb-d object dataset. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 1817–1824. IEEE.

Lai, K., Bo, L., Ren, X., and Fox, D. (2011b). A large-scale hierarchical multi-view rgb-d object dataset. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 1817–1824. IEEE.

Le, Q. V. (2013). Building high-level features using large scale unsupervised learning. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 8595–8598. IEEE.

Liang, D., Weng, K., Wang, C., Liang, G., Chen, H., and Wu, X. (2014). A 3d object recognition and pose estimation system using deep learning method. In *Information Science and Technology (ICIST), 2014 4th IEEE International Conference on*, pages 401–404. IEEE.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157. Ieee.

Nair, V. and Hinton, G. E. (2009a). 3d object recognition with deep belief nets. In *Advances in Neural Information Processing Systems*, pages 1339–1347.

Nair, V. and Hinton, G. E. (2009b). 3d object recognition with deep belief nets. In *Advances in Neural Information Processing Systems*, pages 1339–1347.

Savarese, S. and Fei-Fei, L. (2007). 3d generic object categorization, localization and pose estimation. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE.

Schwarz, M., Schulz, H., and Behnke, S. (2015). Rgb-d object recognition and pose estimation based on pre-trained convolutional neural network features. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 1329–1335. IEEE.

Sermanet, P., Kavukcuoglu, K., Chintala, S., and LeCun, Y. (2013). Pedestrian detection with unsupervised multi-stage feature learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3626–3633.

Smolensky, P. (1986). Information processing in dynamical systems: Foundations of harmony theory.

Socher, R., Huval, B., Bath, B., Manning, C. D., and Ng, A. Y. (2012). Convolutional-recursive deep learning for 3d object classification. In *Advances in Neural Information Processing Systems*, pages 665–673.

Toldo, R., Castellani, U., and Fusiello, A. (2009). A bag of words approach for 3d object categorization. In *Computer Vision/Computer Graphics CollaborationTechniques*, pages 116–127. Springer.

Tombari, F., Salti, S., and Di Stefano, L. (2010). Unique signatures of histograms for local surface description. In *Computer Vision–ECCV 2010*, pages 356–369. Springer.

Tombari, F., Salti, S., and Stefano, L. D. (2011). A combined texture-shape descriptor for enhanced 3d feature matching. In *Image Processing (ICIP), 2011 18th IEEE International Conference on*, pages 809–812. IEEE.

Yu, J., Weng, K., Liang, G., and Xie, G. (2013). A vision-based robotic grasping system using deep learning for 3d object recognition and pose estimation. In *Robotics and Biomimetics (ROBIO), 2013 IEEE International Conference on*, pages 1175–1180. IEEE.