Towards 3D Object Detection with Bimodal Deep Boltzmann Machines over RGBD Imagery

Wei Liu, Rongrong Ji, Shaozi Li
Dep. of Cognitive Science, School of Info. Science and Eng., Xiamen University, China
Fujian Key Lab for Brain-Like Intelligent Systems
{rrji, szlig}@xmu.edu.cn

Abstract

Nowadays, detecting objects in 3D scenes like point clouds has become an emerging challenge with various applications. However, it retains as an open problem due to the deficiency of labeling 3D training data. To deploy an accurate detection algorithm typically resorts to investigating both RGB and depth modalities, which have distinct statistics while correlated with each other. Previous research mainly focus on detecting objects using only one modality, which ignores exploiting the cross-modality cues. In this work, we propose a cross-modality deep learning framework based on deep Boltzmann Machines for 3D Scenes object detection. In particular, we demonstrate that by learning cross-modality feature from RGBD data, it is possible to capture their joint information to reinforce detector trainings in individual modalities. In particular, we slide a 3D detection window in the 3D point cloud to match the exemplar shape, which the lack of training data in 3D domain is conquered via (1) We collect 3D CAD models and 2D positive samples from Internet. (2) adopt pretrained R-CNNs [2] to extract raw feature from both RGB and Depth domains. Experiments on RMRC dataset demonstrate that the bimodal based deep feature learning framework helps 3D scene object detection.

1. Introduction

Coming with the popularities of depth sensors like Kinect, nowadays have witnessed an explosive growth of RGB-Depth (RGBD) data to be processed and analyzed, with extensive applications in robotic navigation, pilotless automobile, gaming and entertainments etc. In the core of such applications lies the problem of RGBD scene parsing, i.e., inferring labels of individual verxels to parse their semantic structure. The parsing follows a similar procedure as what has been done in 2D images. In a typically setting, verxels are first over-segmented into superverxels [1, 4, 5]. Subsequently, semantic labels of individual superverxels are inferred using either discriminative [1, 4] or generative [6, 7] schemes, which are followed by a joint optimization among labels of spatially nearby supervexels using models such as Conditional Random Field (CRF) [1, 4] or Markov Random Filed (MRF) [5, 8].

In this paper, we focus on object detection in RGBD point clouds, which retains as an open problem in the state-of-the-art semantic parsing algorithms of 3D point clouds [6, 7, 9, 10]. Among various designs, detection based labeling has attracted ever increasing research interest [6, 7]. In such a case, detector templates are trained from labeled positive and negative examples for each class labeled offline. However, the detector accuracy heavily depends on the sufficiency of training labels [6, 7], which in turn is very difficult to ensure comparing to the 2D case.

To the best of our knowledge, the existing labels available for RGBD images are mostly hundreds to thousands, for instance, NYU [11], RMRC [10], and SUN3D [12], which is of scales less comparing to endeavors on the image domain like ImageNet [13], LabelMe [14], and Tiny Images [15]. On one hand, as at its earliest stage, it retains a far long way to accumulate sufficient labels from the research community. On the other hand, it is much more difficult to label the RGBD regions or cubics comparing to the 2D case, which typically needs interactive rotation and scaling of the RGBD data rendered in a graphical interface. Furthermore, after transferring the labels into the 3D domain, robust detection algorithms are highly required to handle such a "domain shift".

Therefore, one natural thought is to "transfer" the labels obtained from the 2D domain into the RGBD case to benefit the detector training. The extensive benchmarking works done in 2D domain have accumulated unthinkably rich amount of labels about the objects, containing rich their 2D appearances like shape, texture, and color etc. In the computer vision community, it has led to a decade-long effort in building large-scale labeling datasets, which has shown large benefits for 2D image segmentation, labeling, classi-

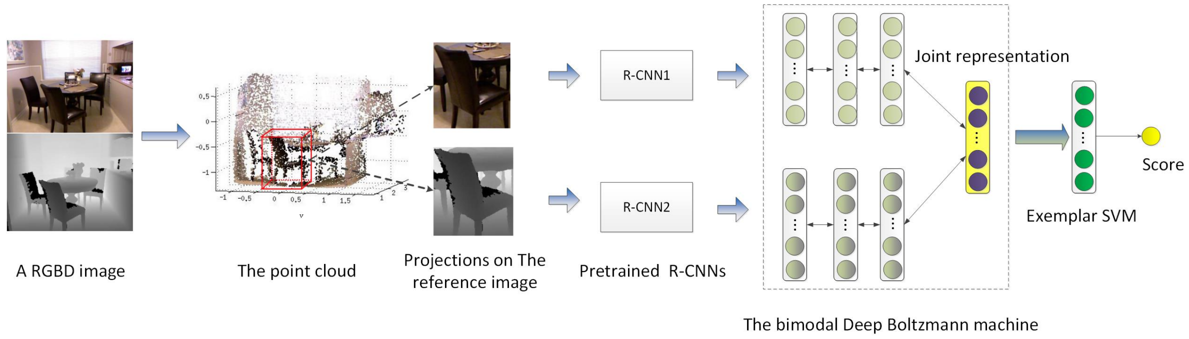

Figure 1. The framework of this paper. To train the bimodal DBM, We utilize labeled 2D samples from existing massive semantic labeled datasets such as ImageNet, and download CAD model from Google Trimble 3D Warehouse to generate 3D positive training samples. During testing time, we sliding exhaustively a bounding box in 3D point cloud to get score for each example of Exemplar-SVMs. To obtain the feature of the 3D bounding box, we first project the 3D bounding box in the 2D bounding boxes on the reference RGBD image. Next, 2D bounding boxes are feeded to pretrained R-CNNs to get raw features for both RGB channel and Depth channel. Then, the bimodal DBM is used to get the joint representation based on the raw features. After that, we are able to get score for each example of Exemplar-SVMs. The detection of score of an Exemplar-SVM indicates whether there is a corresponding shape in the bounding box. At last, non-maximum suppression is performed on all detection boxes in 3D.

fication and object detection. As a compensation, Depth data provides useful spatial information that is possible, but difficult, to estimate from a single RGBD image [17]. Given the deficiency in RGBD data like the one captured from Kinect, e.g., sensor noise, missing depth in different views, as well as occlusions and background clutters [10], it is not doable to directly borrow the labels and data structure from RGB and Depth domains, directly and respectively. However, can we learn feature representations from both RGB data and Depth modalities to benefit the parsing of RGBD data? Such a cross-modality learning, if not impossible, can open a gate to the feature representation design and detector learning for RGBD. And it is so far untouched in the previous research, i.e., previous works [1, 4, 5, 8, 18] mainly focus on learning feature representation and detector using solely one modality.

One existing work comes from Bo et al. [7], which merges the detection results of reference images and the 3D point cloud to improve the parsing accuracy of using single modality. However, the combination is simply a detector score fusion with handcraft weighting, which is unable to describe the complex correlation between both modalities.

In this paper, we conquer this challenge by resorting to a feature-level learning crossing both RGB and Depth modalities. To this end, a bimodal deep learning framework is proposed to learn robust detectors in RGBD domain, as shown in the framework in Figure 1. Our innovation is two-fold: For the bimodal feature learning, we utilize deep Boltzmann Machine(DBM) to learn features over RGB data and Depth data. For the robust detection, we train Exemplar-SVMs using fused representations of the learned DBM, it ensures the flexibility and generality by training instance-specific metrics and classifiers.

The rest of this paper is organized as follows. Section 2 surveys related work. Section 3 introduces the proposed scheme. The detailed bimodal DBM is provided in Section 4. Section 5 illustrates the training of Exemplar-SVMs. Experimental results and comparison with existing methods are provided in Section 6. Finally, we conclude this paper in Section 7.

2. Related Work

In the literature, a significant amount of works has been done for object detection in 3D scenes. A considerable proportion of these works focus on feature design. For instance, the works proposed in [1, 4, 16, 19] employed features such as gDPM. The works proposed in [10] and [7] utilized Sparse coding to learn feature representations. Wang et al. [9] employed unsupervised feature from raw RGBD input with two-layer stacking structure. Features of the two layers are concatenated to train linear SVMs over superpixels for semantic labeling. The works in [16, 21] first parse over 2D images, and then project classification results from reference RGBD images into 3D point cloud. The works in [1, 4, 5]first oversegment 3D point cloud into supervoxels, and then jointly label such segments by CRFs. Shrivastava and Gupta [19] proposed part-based representation for modeling object categories.

To overcome the challenge of limited training data, several recent works [7, 10, 16] proposed to conduct label transfer from related domains, such as LabelMe and ImageNet. For example, the works in [7] and [10] leveraged CAD models from Google Trimble 3D Warehouse to render 3D data of targeted objects as positive samples. Wang et al. [16] employed the existing massive 2D semantic labeled datasets, such as ImageNet and LabelMe, in combination with Exemplar-SVMs [23] based classifier for label transfer from 2D images to 3D point clouds. However, so far none existing works exploits the joint learning between 2D and 3D in terms of the feature representation part.

In recent years, deep learning techniques have been successfully applied to learn cross-modality features [24, 25]. Ngiam et al. [24] have proposed a deep learning based multi-modality learning scheme that has shown to outperform the features learned from single modality. Srivastava and Salkhutdinov[25] proposed a DBM model for learning a generative model of data that consists of multiple and diverse input modalities. The model works by learning a probability over the space of visible units, in which states of latent variables are leveraged as joint representation of multi-modality input.

The works that are most similar to our work are [7] and [10]. There are, however, some fundamental differences. First, in this work, we collect both 2D and Computer Graphics(CG) CAD models training data from Internet. Second, we focus on among two diverse modalities: RGB and Depth. Third, to cope with the challenge of deficiency of training data, pretrained R-CNNs are used to extract raw feature from both RGB and Depth channels.

3. The Proposed Framework

Given a RGBD image of a scene, the detection task is to find instances of real-world objects such as *chair* and *table*, which are represented as 3D cuboids. Figure 1 presents an overview of the proposed framework. Our framework takes a RGBD image from Kinect with the gravity direction as input. Most objects are assumed to be aligned on gravity direction so there is only rotation around gravity axis. To support 3D sliding window, the 3D space is divided into cubic cells of size 0.1 meter. For online detection, given a RGBD image, we first generate a point cloud of the scene, based on cameral parameters [10]. Next, we exhaustively slide a 3D bounding box in the point cloud to get scores for all Exemplar-SVMs. Then, the 3D bounding box is projected into 2D bounding boxs on RGB channel and Depth channel, with the reference RGBD image. After that, raw features and the fused representation are sequentially extracted by R-CNNs and the proposed bimodal DBM respectively. Then, the joint representation is used to get scores for all Exemplar-SVMs. Finally, non-maximum suppression is performed on all detection boxes in 3D. The process of detecting objects of interest is summarized in Algorithm 1.

3.1. Generating Positive Training Samples

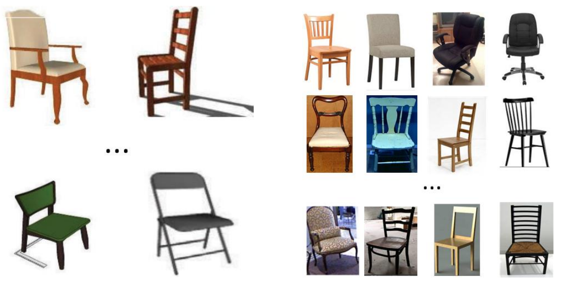
(a) CAD models (b) 2D positive training data

Figure 2. Positive training samples for *chair* leveraged from Internet:(a) CAD models from Google Trimble 3D Warehouse. (b) RGB positive training samples cropped form images leveraged from Google and ImageNet.

As shown in Figure 2, for each object category, we retrieve positive training data using the corresponding keyword. Concretely, to generate 3D cuboids as positive training examples, we follow [10] to make use of 3D CAD models collected from Trimble 3D Warehouse. Each CG CAD model is rendered from different viewing angles and locations in the 3D space to obtain synthetic depth maps, as if they are viewed by a typical RGBD sensor.[1] To handle shape and viewpoint variance, each CG model is densely rendered by varying the following parameters: orientation, scale, 3D location, and camera tilt angle. For each rendering, we train an Exemplar-SVM model. For each category, all SVMs from rendering of CG models are assembled to build a 3D detector.

An important observation is that amount of object instances for each category in RMRC dataset is small. However, the sufficiency of positive samples is a prerequisite to train a reliable DBM. To address this issue, we augment the positive RGB samples by retrieving from Internet. More specially, to generate 2D positive training samples for a given category such as chair, we first retrieve images of the target categories from ImageNet and Google, using the corresponding keyword, after that positive samples are cropped from images.

3.2. Generating Negative Samples

For a category, we train Exemplar-SVMs model [23]. And all SVMs from renderings of CG chair models are assembled to build a 3D chair detector. RGB and Depth raw features are extracted to feed into our bimodal DBM. Then the fused representations are utilized to train the SVM. Arise the fact that the training performance heavily depends on how to collect rich negative examples, hard negative mining is performed in the phrase of training. The initial negative samples are randomly picked from annotated RGBD images in RMRC dataset [26] that do not overlap with ground truth positives. After training the detector, hard

[1] In rendering, we use the same camera intrinsic parameters, and resolution to virtual camera

Algorithm 1 Object detection for RGBD images

```
Input:
    A RGBD image with the gravity direction, $I$
    Cameral parameters of the RGBD image, $p$
    The Learned bimodal DBM
    An ensemble of trained exemplar-SVMs,$\mathbb{E}$
    Object instances, $\mathcal{O} = \varnothing$
Output:
    $\mathcal{O}$
Construct a point cloud $c$ for image $I$ basing on $P$
for each Exemplar-SVM classifier $e \in \mathbb{E}$ do
    Slide a 3D bounding box in $c$
    for each 3D bounding box do
        Get projections on RGBD reference image
        Extract $f_m$ and $f_d$ from RGB and Depth modalities respectively, using R-CNNs.
        Feed $f_m$ and $f_d$ into the bimodal DBM and obtain a joint representation $f$
        Classify $f$ with $e$ and then get a score $s$ for the 3D bounding box
    end for
end for
Perform non-maximum suppression on all detection boxes
```

negative mining is performed by searching hard negatives over the entire training set.

4. Bimodal Feature Learning

We adopt bimodal Boltzmann deep machines for bimodal feature learning. As shown in Figure 3, a bimodal deep Boltzmann machine is constructed with two DBMs by adding an additional layer of binary hidden units on top of them. In this section we first review the Restricted Boltzmann machines(RBMs), and then describe the proposed bimodal Boltzmann deep machines in detail.

4.1. Restricted Boltzmann Machines

Boltzmann machines(BMs) are a particular form of log-linear Markov Random Field, with stochastic visible units $\mathbf{v} \in \{0,1\}^D$ and stochastic hidden units $\mathbf{h} \in \{0,1\}^F$. RBMs further restrict $\mathbf{v}$ and $\mathbf{h}$ of BMs to be two disjoint sets, such that each visible unit connected to each hidden unit. The energy function over the visible and hidden units $\{0,1\}^{D+F}$ of an RBM is defined as:

$$E(\mathbf{v}, \mathbf{h}; \theta) = -\mathbf{h}^{'}\mathbf{W}\mathbf{v} - \mathbf{b}^{'}\mathbf{v} - \mathbf{a}^{'}\mathbf{h}, \tag{1}$$

where $\theta = \{\mathbf{a}, \mathbf{b}, \mathbf{W}\}$ are the model parameters. The joint distribution of the energy-based probabilistic model is defined through an energy function as follows:

$$P(\mathbf{v}, \mathbf{h}; \theta) = \frac{1}{\mathcal{Z}(\theta)} exp(-E(\mathbf{v}, \mathbf{h}; \theta)), \tag{2}$$

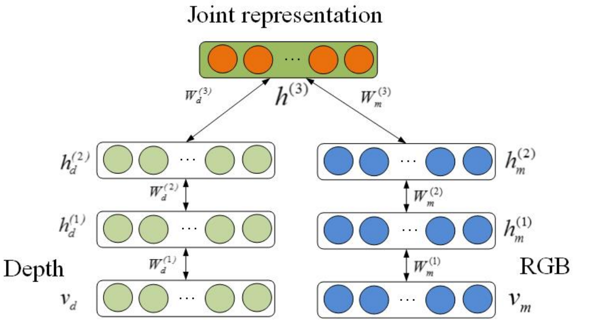

Figure 3. Bimodal DBM. A bimodal DBM modeling the joint representation over RGB data and Depth data intputs.

in which the normalizing factor $\mathcal{Z}(\theta)$ is called the partition function.

Consider modeling Gaussian-Bernoulli RBMs[20, 22], that is, let $\mathbf{v} \in \mathbb{R}^D$ be real-valued Gaussian variables, and $\mathbf{h} \in \{0,1\}^F$ be binary stochastic hidden units. The energy of Gaussian-Bernoulli RBM over $\{\mathbf{v}, \mathbf{h}\}$ is defined as:

$$E(\mathbf{v}, \mathbf{h}; \theta) = \sum_{i=1}^{D} \frac{(v_i - b_i)^2}{2\sigma_i^2} - \sum_{i=1}^{D} \sum_{j=2}^{F} \frac{v_i}{\sigma_i} W_{ij} h_j - \sum_{j=1}^{F} a_j h_j. \tag{3}$$

4.2. RGB-Depth DBM

A DBM [27] is a network of symmetrically coupled stochastic binary units. It contains a set of visible units $\mathbf{v} \in \{0,1\}^D$, and a sequence of layers of hidden units $\mathbf{h}^{(1)} \in \{0,1\}^{F_1}$, $\mathbf{h}^{(2)} \in \{0,1\}^{F_2}$, ..., $\mathbf{h}^{(L)} \in \{0,1\}^{F_L}$. Connections only exist between hidden units in adjacent layers, as well as between visible and hidden units in the first hidden layer.

Consider modeling a RGB-specific Gaussian-Bernoulli DBM with three hidden layers, let $\mathbf{v}^m \in \mathbb{R}^D$ denote a real-valued image input. Let $\mathbf{h}^{(1m)} \in \{0,1\}^{F_1^m}$, $\mathbf{h}^{(2m)} \in \{0,1\}^{F_2^m}$, and $\mathbf{h}^{(3m)} \in \{0,1\}^{F_3^m}$ be the three layers of hidden units in the RGB-specific DBM. Then energy of Gaussian-Bernoulli DBM over $\{\mathbf{v}^m, \mathbf{h}^m\}$ is defined as:

$$\begin{aligned} E(\mathbf{v}^m, \mathbf{h}^m; \theta^m) = & -\sum_{i=1}^{D} \sum_{j=1}^{F_1^m} \frac{v_i^{(m)}}{\sigma_i^{(m)}} W_{ij}^{(1m)} h_j^{(1m)} \\ & - \sum_{j=1}^{F_1^m} \sum_{l=1}^{F_2^m} W_{jl}^{(2m)} h_j^{(1m)} h_l^{(2m)} - \sum_{l=1}^{F_2^m} \sum_{p=1}^{F_3^m} W_{lp}^{(3m)} h_l^{(2m)} h_p^{(3m)} \\ & + \sum_{i=1}^{D} \frac{(v_i^{(m)} - b_i^{(m)})^2}{2{\sigma_i^{(m)}}^2} - \sum_{j=1}^{F_1^m} b_j^{(1m)} h_j^{(1m)} - \sum_{l=1}^{F_2^m} b_l^{(2m)} h_l^{(2m)} \\ & - \sum_{p=1}^{F_3^m} b_p^{(3m)} h_p^{(3m)}, \end{aligned} \tag{4}$$

where σ_i is deviation of the corresponding Gaussian model, and θ^m is the parameter vector of DBM. Therefore, the joint distribution of the energy-based probabilistic model is defined through an energy function as:

$$P(\mathbf{v}^m; \theta^m) = \frac{1}{Z(\theta^m)} \sum_{\mathbf{h}^m} \exp\big(- E(\mathbf{v}^m, \mathbf{h}^m; \theta^m)\big), \quad (5)$$

where $Z(\theta^m)$ is the partition function. Similarly, the corresponding probability assigned to $\mathbf{v}^d$ by Depth-specific DBM has the same form with Equation 5. We model our Image-Depth DBM by two Gaussian-Bernoulli DBMs, as shown in Figure 3. The proposed bimodal DBM is constructed with two DBMs by adding an additional layer of binary hidden units on top of them. let $\mathbf{v}^m \in \mathbb{R}^D$ and $\mathbf{v}^D \in \mathbb{R}^K$ denote a real-valued RGB input and a real-valued Depth input respectively. Consider modeling an Image-Depth DBM with three hidden layers, let $\{\mathbf{v}^m, \mathbf{v}^d\}$ be real-valued Gaussian variables, and $\{\mathbf{h}^{(1m)}, \mathbf{h}^{(2m)}, \mathbf{h}^{(1c)}, \mathbf{h}^{(2c)}, \mathbf{h}^{(3)}\}$ be binary stochastic hidden units. Let $\mathbf{h}^{(1m)} \in \{0,1\}^{F_1^m}$ and $\mathbf{h}^{(2m)} \in \{0,1\}^{F_2^m}$ be the two layers of hidden units in the RGB-specific two layer DBM. Similarly, let $\mathbf{h}^{(1d)} \in \{0,1\}^{F_1^c}$ and $\mathbf{h}^{(2d)} \in \{0,1\}^{F_2^c}$ be the two layers of hidden units in the cloud-specific two layer DBM. The energy of the proposed bimodal Gaussian-Bernoulli DBM over $\{\mathbf{v}, \mathbf{h}\}$ can be defined as:

$$\begin{aligned}
E(\mathbf{v}, \mathbf{h}; \theta) = &- \sum_{i=1}^{D} \sum_{j=1}^{F_1^m} \frac{v_i^{(m)}}{\sigma_i^{(m)}} W_{ij}^{(1m)} h_j^{(1m)} \\
&- \sum_{j=1}^{F_1^m} \sum_{l=1}^{F_2^m} W_{jl}^{(2m)} h_j^{(1m)} h_l^{(2m)} - \sum_{l=1}^{F_2^m} \sum_{p=1}^{F_3} W_{lp}^{(3m)} h_l^{(2m)} h_p^{(3)} \\
&+ \sum_{i=1}^{D} \frac{(v_i^{(m)} - b_i^{(m)})^2}{2{\sigma_i^{(m)}}^2} - \sum_{j=1}^{F_1^m} b_j^{(1m)} h_j^{(1m)} - \sum_{l=1}^{F_2^m} b_l^{(2m)} h_l^{(2m)} \\
&- \sum_{i=1}^{K} \sum_{j=1}^{F_1^d} \frac{v_i^{(d)}}{\sigma_i^{(d)}} W_{ij}^{(1d)} h_j^{(1d)} \\
&- \sum_{j=1}^{F_1^d} \sum_{l=1}^{F_2^d} W_{jl}^{(2d)} h_j^{(1d)} h_l^{(2d)} - \sum_{l=1}^{F_2^d} \sum_{p=1}^{F_3} W_{lp}^{(3c)} h_l^{(2d)} h_p^{(3)} \\
&+ \sum_{i=1}^{K} \frac{(v_i^{(d)} - b_i^{(d)})^2}{2{\sigma_i^{(d)}}^2} - \sum_{j=1}^{F_1^d} b_j^{(1d)} h_j^{(1d)} - \sum_{l=1}^{F_2^d} b_l^{(2d)} h_l^{(2d)} \\
&- \sum_{p=1}^{F_3} b_p^{(3)} h_p^{(3)}.
\end{aligned} \quad (6)$$

Therefore, the joint probability distribution over the bimodal input $\{\mathbf{v}^m, \mathbf{v}^d\}$ can be written as:

$$\begin{aligned}
P(\mathbf{v}^m, \mathbf{v}^c; \theta) = &\sum_{\boldsymbol{h}^{(2m)}, \boldsymbol{h}^{(2d)}, \boldsymbol{h}^{(3)}} P(\boldsymbol{h}^{(2m)}, \boldsymbol{h}^{(2d)}, \boldsymbol{h}^{(3)}) \\
&\Big(\sum_{\boldsymbol{h}^{(1m)}} P(\boldsymbol{v}^m, \boldsymbol{h}^{(1m)}, \boldsymbol{h}^{(2m)}) \Big) \Big(\sum_{\boldsymbol{h}^{(1d)}} P(\boldsymbol{v}^d, \boldsymbol{h}^{(1d)}, \boldsymbol{h}^{(2d)}) \Big) \\
= &\frac{1}{\mathcal{Z}(\theta)} (\sum_{\mathbf{h}} exp\Big(- \sum_i \frac{(v_i^m)^2}{2\sigma_i^2} + \\
&\sum_{ij} \frac{v_i^m}{\sigma_i} W_{ij}^{(1m)} h_j^{(1m)} + \sum_{jl} W_{jl}^{(2m)} h_j^{(1m)} h_l^{(2m)} \\
&- \sum_i \frac{(v_i^d)^2}{2\sigma_i^2} + \sum_{ij} \frac{v_i^d}{\sigma_i} W_{ij}^{(1d)} h_j^{(1d)} + \sum_{jl} W_{jl}^{(2d)} h_j^{(1d)} h_l^{(2d)} \\
&+ \sum_{lp} W^{(3m)} h_l^{(2m)} h_p^{(3)} + \sum_{lp} W^{(3c)} h_l^{(2d)} h_p^{(3)} \Big).
\end{aligned} \quad (7)$$

The task of learning the bimodal DBM is the maximum likelihood learning for Equation 7 respect to the model parameters.

4.3. Approximate Inference And Learning

Though exact maximum likelihood learning in the bimodal DBM is intractable, but there exists a good stochastic approximate learning [27] carried out by using mean-field inference and a MCMC based stochastic approximation. Specifically, during the inference step, the true posterior $P(\mathbf{h}|\mathbf{v}; \theta)$ is approximated with a fully factorized approximating distribution over the five sets of hidden units $\{\mathbf{h}^{(1m)}, \mathbf{h}^{(2m)}, \mathbf{h}^{(1d)}, \mathbf{h}^{(2d)}, \mathbf{h}^{(3)}\}$:

$$\begin{aligned}
Q(\mathbf{h}|\mathbf{v}; \boldsymbol{\mu}) = &\Big(\prod_{j=1}^{F_1^m} q(h_j^{(1m)}|\mathbf{v}) \prod_{l=1}^{F_2^m} q(h_l^{(2m)}|\mathbf{v}) \Big) \\
&\Big(\prod_{j=1}^{F_1^d} q(h_j^{(1d)}|\mathbf{v}) \prod_{l=1}^{F_2^d} q(h_l^{(2d)}|\mathbf{v}) \Big) \Big(\prod_{k=1}^{F_3} q(h_k^{(3)}|\mathbf{v}) \Big),
\end{aligned} \quad (8)$$

where $\boldsymbol{\mu} = \{\boldsymbol{\mu}^{(1m)}, \boldsymbol{\mu}^{(2m)}, \boldsymbol{\mu}^{(1d)}, \boldsymbol{\mu}^{(2d)}, \boldsymbol{\mu}^{(3)}\}$ are the mean-field parameters with $q(h_i^{(l)} = 1|\mathbf{v}) = \mu_i^{(l)}$ for $l = 1, 2, 3$.

For each training example, learning proceeds as follows. First, a greedy layer-wise pretraining strategy by learning a stack of modified RBMs is employed to initialize the model parameters. Then by finding the value of the variational parameters $\boldsymbol{\mu}$ that maximizes the variational lower bound for the current fixed model parameters θ. After that, given the value of $\boldsymbol{\mu}$, we update the model parameters using a MCMC based stochastic approximation [27, 25].

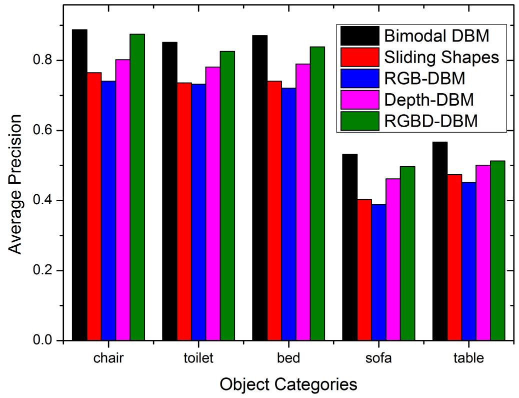

Figure 4. Average precision for various algorithms, object categories on normal ground truth boxes.

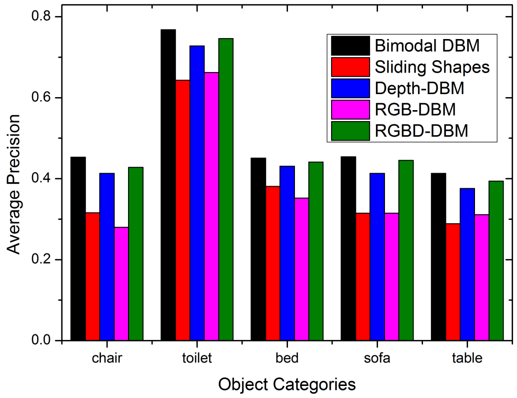

Figure 5. Average precision for various algorithms, object categories on all ground truth boxes including difficult cases

5. Training Exemplar SVMs

The process of training a Exemplar-SVM is demonstrated in Figure 7. For each of the synthetic depth maps, we extract a joint representation and take it as the single positive sample to train an Exemplar-SVM. Initially, many negatives are randomly picked point cloud from labeled RMRC dataset not overlapped with ground truth positives. Hard negative mining is performed by searching hard negatives over the entire training set. It is worth to note that, for each of the synthetic depth maps we could only extract raw features from the Depth channel. Yet, the fused representation of the positive is able to be inferred by the designed bimodal DBM when RGB raw features is missing.

6. Experimental validations

6.1. Experiment setup

Data set: We evaluate our 3D detector on RMRC dataset [26]. The RMRC dataset is a data set for indoor challenges, taken from the NYU Depth V2 dataset. The 3D detection task contains 1074 RGB and Depth frames. The classes to detect include: *bed*, *table*, *sofa*, *chair*, *television*, *desk*, *toilet*, *monitor*, *dresser*, *night stand*, *pillow*, *box*, *door*, *garbage bin*, and *bathtub*. Each image has been annotated with 3D bounding boxes. We select five objects: *chair*, *bed*, *toilet*, *sofa*, and *table*, which are well labeled in the dataset. For comparison, the training-test splits in our experiments are set to be the same with [10]. The RMRC dataset is partitioned into 500 depth images for training and 574 depth images for testing, in a way that the images from same video are grouped together and appear only in training or testing set. The instance number are balanced in training and testing set for each category.

Raw features: To support sliding a 3D bounding box in a point cloud, we first divide the point cloud into cells of 0.1 meter. During testing and hard-negative mining, we slide a 3D detection window in 3D space generated by a RGBD image. To extract the raw features of a 3D detection window from both RGB domain and Depth domain, two pretrained R-CNNs [2] corresponding to RGB channel and HHA channel respectively are leveraged to extract features from projections on the RGBD reference image. Both of the two R-CNNs have about 60 million parameters and were first pretrained on approximately 1.2 million RGB images from the ILSVRC 2012 dataset [3] and then finetuned on a much smaller detection dataset [2]. We compute features at the fully connected layer 6 from both of the R-CNNs. Two 4,096 dimensional features are obtained from two diverse modalities.

Model architecture: Both the RGB pathway and Depth pathway consists of Gaussian RBM with 4,096 visible units and 1,024 hidden units, followed by a layer of 1,024 binary hidden units. The Joint layer consists of 2,048 binary hidden units. All of the Gaussian visible units are set to have unit variance. Each dimension of feature is standardized to have zero-mean and unit-variance, before feeded to Bimdal DBM.

Comparison Baselines: We quantitatively evaluate the performance of our scheme with the following baselines: (1) Sliding Shapes [10]. (2) Depth-DBM: We trained a DBM using only raw Depth features. And during testing time, the model was given only the Depth inputs. (3) RGB-DBM: We trained a DBM using only raw RGB features. And during testing time, the model was given only the RGB inputs. (4) RGBD-DBM: concatenated representations of Depth-DBM and image RGB-DBM that are trained separately.

Evaluation metric: we adopt the 3D bounding box overlapping ratio [10] and Mean Average Precision (MAP) to evaluate the performances of the schemes. A predict box

Model	MAP	
	3D+	3D
Bimodal DBM	**0.508**	**0.742**
Sliding Shapes	0.389	0.624
Depth-DBM	0.442	0.667
RGB-DBM	0.384	0.607
RGBD-DBM	0.491	0.718

Table 1. Mean average precision for various algorithms on the RMRC dataset.

is considered to be correct if the overlapping ratio is more than 0.25. we also add a difficult flag to indicate whether the ground truth is difficult to detect. The difficult cases include heavy occlusion, missing depth and out of sight. We evaluate on normal ground truth boxes (denoted as 3D), as well as on all ground truth boxes including difficult cases (denoted as 3D+) respectively. The MAP is the mean average precision for 5 object categories.

6.2. Experimental Results For Baselines

In Figure 4 and Figure 5, we show that the proposed method yields better performance than the compared methods. As shown in Table 1, on normal ground truth boxes the proposed bimodal DBM achieves MAP of 0.508, compared to 0.389, 0.472, 0.384 and 0.491, achieved by Sliding Shapes, Depth-DBM, RGB-DBM, Sliding Shapes, and RGBD-DBM, respectively. And on all ground truth boxes including difficult cases, the proposed bimodal DBM achieves MAP of 0.742, compared to 0.624, 0.667, 0.607 and 0.718, achieved by Sliding Shapes, Depth-DBM, RGB-DBM, Sliding Shapes, and RGBD-DBM, respectively. The above comparisons show that RGB data and Depth data are in complementary to each other for object detection in RGBD scenes. The fused representation of features from the two diverse modalities is able to boost the performance of object detection in RGBD scenes. In particular, the bimodal DBM is good at discovering useful fused representation for data from the two diverse modalities.

6.3. Insights Of the Bimodal DBM

In Figure 6, we compare the performance using representation from different layers of bimodal DBMs. We do this by measuring average precision obtained by Exemplar-SVMs trained on the representation at different layers of the bimodal DBM. Each layer of the bimodal DBM provides a different representation of the input. The input layers, as shown in the bottom ends, are the raw features. Figure 6 demonstrates that, as we go deeper into the model from the input layer to middle layer, the internal representation get better. In particular, the joint representation of two modal serves as the best useful feature representation. Therefore, this experiment result further shows that our bimodal DBM is able to learn useful unified representation for the task of object detection with RGBD images.

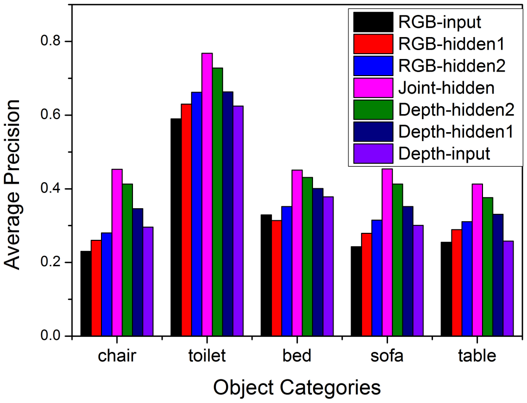

Figure 6. Average precision using representation from different layers of bimodal DBMs.

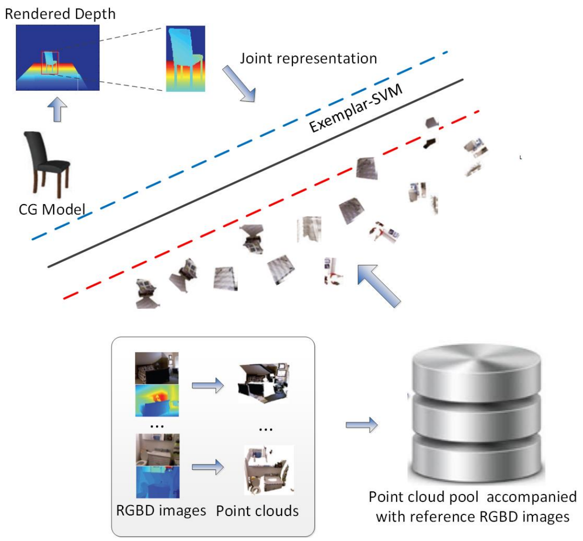

Figure 7. **An Ensemble of Exemplar-SVMs**. We train a separate linear SVM for each of the synthetic depth maps. To get fused representations, RGB and Depth raw features are extracted to feed into our bimodal DBM. Then the fused representations are utilized to train the SVM.

6.4. On Transferring 2D Labels From Internet

In this experiment, we evaluate the ability of the bimodal model to improve detection by cropping 2D positive training data from Internet. Both models have the same depth and the same number of hidden units in each layer. By this setting, we can assess the contribution of the 2D positive training data transferred from Internet. Figure 8 and Figure 9 demonstrate that bimodal DBM with 2D positive training data performs better than the compared baselines.

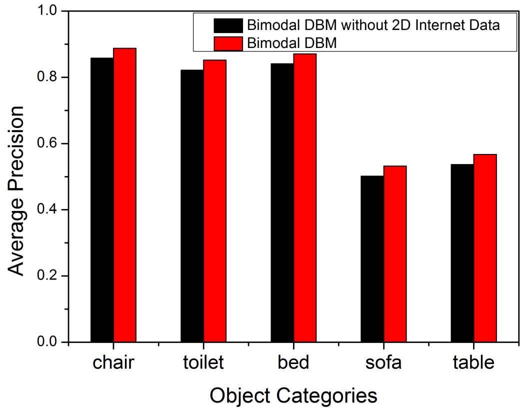

Figure 8. Average precision on normal ground truth boxes.

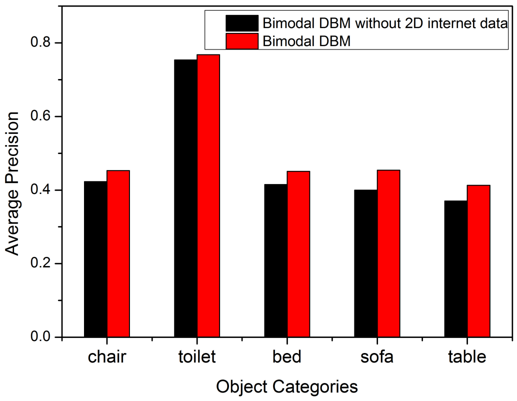

Figure 9. Average precision on all ground truth boxes including difficult cases.

This experiment shows that the 2D positive training data cropped from cross domain can improve object detection accuracy for RGBD images to some extend.

6.5. Computation Cost Analysis

We spend most of the time on the training of the bimodal DBM and Exemplar SVMs. Though the number of parameters of the bimodal DBM reach millions, the parameters are able to be well trained by using GPUs [25]. With NVIDIA GTX680, we can train a bimodal DBM within two days for the category of *chair*. For training Exemplar SVMs, it takes about 10 hours to train a single detector with all its exemplar SVMs with single thread in Matlab For testing, it takes about 5 second per detector to test on a RGBD image in Matlab. During the process of testing, to speed up the process, jumping window [10] is utilized to skip empty space in 3D point without loss of detection precision and recall.

7. Conclusion

3D object detection in RGBD scenes is a very challenging task due to the deficiency of training data. To conquer this challenge, in this paper we propose a deep feature learning framework based on deep Boltzmann Machines in combination with Exemplar-SVMs based robust detector. The experiments on the RMRC dataset demonstrate the superiority of our framework. In our future work, we will integrate our scheme for multi-task structure learning to exploit the context information among 3D scenes to further improve the accuracy and robustness of the proposed 3D object detectors.

8. Acknowledgement

This work is supported by the Excellent Youth Science Foundation of NSFC (No. 61422210), and the Nature Science Foundation of China (No. 61373076).

References

[1] H. S. Koppula, A. Anand, T. Joachims, and A. Saxena. Semantic labeling of 3d point clouds for indoor scenes. In *Advances in Neural Information Processing Systems*, pages 244–252, 2011. 1, 2

[2] S.Gupta, R.Girshick R, P.Arbelez, and J Malik. Learning rich features from RGB-D images for object detection and segmentation. In *European Conference on Computer Vision*,pages 4: 345-360,2014. 1, 6

[3] Deng J, Berg A, Satheesh S, et al. ImageNet large scale visual recognition competition 2012(ILSVRC2012). 2012. 6

[4] O. Kahler and I. Reid. Efficient 3d scene labeling using fields of trees. In *International Conference on Computer Vision*, pages 3064–3071, 2013. 1, 2

[5] M. Najafi, S. T. Namin, M. Salzmann, and L. Petersson. Non-associative higher-order markov networks for point cloud classification. In *European Conference on Computer Vision*, pages 500–515, 2014. 1, 2

[6] K. Lai, L. Bo, X. Ren, and D. Fox. Detection-based object labeling in 3d scenes. In *IEEE International Conference on Robotics and Automation*, pages 1330–1337, 2012. 1

[7] L. Bo, X. Ren, and D. Fox. Unsupervised feature learning for rgb-d based object recognition. In *Experimental Robotics*, pages 387–402, 2013. 1, 2, 3

[8] X. Xiong, D. Munoz, J. A. Bagnell, and M. Hebert. 3-d scene analysis via sequenced predictions over points and regions. In *IEEE International Conference on Robotics and Automation*, pages 2609–2616, 2011. 1, 2

[9] A. Wang, J. Lu, G. Wang, J. Cai, and T.-J. Cham. Multi-modal unsupervised feature learning for rgb-d scene labeling. In *European Conference on Computer Vision*, pages 453–467. Springer, 2014. 1, 2

[10] S. Song and J. Xiao. Sliding shapes for 3d object detection in rgb-d images. In *European Conference on Computer Vision*, 2014. 1, 2, 3, 6, 8

[11] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision*, pages 746–760. 2012. 1

[12] J. Xiao, A. Owens, and A. Torralba. Sun3d: A database of big spaces reconstructed using sfm and object labels. In *International Conference on Computer Vision*, pages 1625–1632, 2013. 1

[13] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition*, pages 248–255. 1

[14] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: a database and web-based tool for image annotation. *International journal of computer vision*. 1

[15] A. Torralba, R. Fergus, and W. T. Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 1

[16] Y. Wang, R. Ji, and S.-F. Chang. Label propagation from imagenet to 3d point clouds. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3135–3142, 2013. 2, 3

[17] A. Saxena, M. Sun, and A. Y. Ng. Make3d: Learning 3d scene structure from a single still image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 824–840, 2009. 2

[18] A. Anand, H. S. Koppula, T. Joachims, and A. Saxena. Contextually guided semantic labeling and search for three-dimensional point clouds. *The International Journal of Robotics Research*, pages 1–16, 2012. 2

[19] A. Shrivastava and A. Gupta. Building part-based object detectors via 3d geometry. In *IEEE International Conference on Computer Vision*, pages 1745–1752, 2013. 2

[20] Y. Freund and D. Haussler. *Unsupervised learning of distributions of binary vectors using two layer networks*. Computer Research Laboratory University of California, Santa Cruz, 1994. 4

[21] C. Herbon, B. Otte, K. Tönnies, and B. Stock. Detection of clustered objects in sparse point clouds through 2d classification and quadric filtering. In *Pattern Recognition*, pages 535–546. 2014. 2

[22] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*. 4

[23] T. Malisiewicz, A. Gupta, and A. A. Efros. Ensemble of exemplar-svms for object detection and beyond. In *IEEE International Conference on Computer Vision*, pages 89–96, 2011. 3

[24] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng. Multimodal deep learning. In *Proceedings of the 28th International Conference on Machine Learning*, pages 689–696, 2011. 3

[25] N. Srivastava and R. Salakhutdinov. Multimodal learning with deep boltzmann machines. In *Advances in neural information processing systems*, pages 2222–2230, 2012. 3, 5, 8

[26] R. Guo and D. Hoiem. Support surface prediction in indoor scenes. In *IEEE International Conference on Computer Vision*, pages 2144–2151, 2013. 3, 6

[27] R. Salakhutdinov and G. E. Hinton. Deep boltzmann machines. In *International Conference on Artificial Intelligence and Statistics*, pages 448–455, 2009. 4, 5