2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Macau, China, November 4-8, 2019

Grounding Language Attributes to Objects using Bayesian Eigenobjects

Vanya Cohen*, Benjamin Burchfiel*, Thao Nguyen*, Nakul Gopalan, Stefanie Tellex, George Konidaris
vanya_cohen@alumni.brown.edu, bcburch@cs.duke.edu,
{thao_nguyen3, nakul_gopalan, stefanie_tellex, george_konidaris}@brown.edu
* Authors contributed equally

Abstract— **We develop a system to disambiguate object instances within the same class based on simple physical descriptions. The system takes as input a natural language phrase and a depth image containing a segmented object and predicts how similar the observed object is to the object described by the phrase. Our system is designed to learn from only a small amount of human-labeled language data and generalize to viewpoints not represented in the language-annotated depth image training set. By decoupling 3D shape representation from language representation, this method is able to ground language to novel objects using a small amount of language-annotated depth-data and a larger corpus of unlabeled 3D object meshes, even when these objects are partially observed from unusual viewpoints. Our system is able to disambiguate between novel objects, observed via depth images, based on natural language descriptions. Our method also enables viewpoint transfer; trained on human-annotated data on a small set of depth images captured from frontal viewpoints, our system successfully predicted object attributes from rear views despite having no such depth images in its training set. Finally, we demonstrate our approach on a Baxter robot, enabling it to pick specific objects based on human-provided natural language descriptions.**

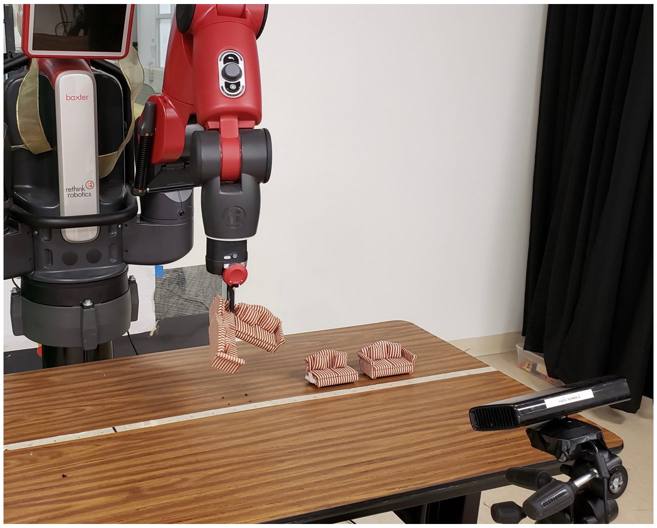

Fig. 1. Our pipeline receives depth images and a natural language command such as, "Pick up the bent couch", to retrieve the described couch.

I. Introduction

As robots grow increasingly capable of understanding and interacting with objects in their environments, a key bottleneck to widespread robot deployment in human-centric environments is the ability for non-experts to communicate with robots. One of the most sought after communication modalities is natural language, allowing a non-expert user to verbally issue directives. We focus our work here, applying natural language to the task of object-specification—describing which of many objects is being referred to by a user, is critically important when tasking a robot to perform actions such as picking, placing, or retrieving an item. Even simple commands, such as *"bring me a small espresso cup"*, require object-specification to be successful.

Object-specification becomes far more difficult when the robot is forced to disambiguate between multiple objects of the same type (such as between several different teapots, rather than between a teapot and a bowl). The solution to this problem is to include descriptive language to not only specify object types, but also object attributes such as shape, size, or more abstract features. Many such descriptions are largely physical in nature, relating directly or indirectly to the actual shape of an object. For instance, descriptions like "round", "tall", "vintage", and "retro", all relate to shape. While current work exists to ground images to natural language [1], [2], these systems only work when an object is observed from a similar viewpoint to human-annotated training images. We must also handle partially observed objects; it is unreasonable for a robot to observe objects in its environment from many angles with no self-occlusion. Furthermore, as human-annotated language data is expensive to gather, it is important to not require language annotated depth images from all possible viewpoints. A robust system must be able to generalize from a small human-labeled set of (depth) images to novel and unique object views. We develop a system that addresses these challenges by grounding descriptions to objects via explicit reasoning over 3D structure. We combine Bayesian Eigenobjects (BEOs) [3], a framework which enables estimation of a novel object's full 3D geometry from a single partial view, with a language grounding model. This combined model uses BEOs to predict a low-dimensional object embedding into a pose-invariant learned object space and predicts language grounding from this low-dimensional space. Critically, BEOs are well suited to the novel object case; using previously seen objects they generalize well to make predictions about previously unseen objects, even from partial views.

This structure has several advantages. First, it enables training the object-understanding portion of the system from a large set of non-language-annotated objects, reducing the need for expensive human-generated object attribute labels

978-1-7281-4004-9/19/$31.00 ©2019 IEEE

to be obtained for all the training data. Second, because the language model learns to predict language groundings from a low-dimensional subspace instead of high-dimensional 2.5D or 3D input, the model complexity, and amount of labeled training data required, is small; in our experiments, a simple 3-layer network was sufficient. Finally, because BEOs are already well suited to novel objects, our system scales well to new objects and partial views, an important capability for a robot operating in an unstructured human-centric environment. Unlike a single monolithic system which would require human-annotated depth images from all possible viewpoints, our approach allows a small number of annotated depth images from a limited set of viewpoints to generalize to significantly novel partial views and novel objects.

We evaluate our system on a dataset of several thousand ShapeNet object instances [4] across three object classes, paired with human-generated object descriptions obtained from Amazon Mechanical Turk (AMT). We show that not only is our system able to distinguish between objects of the same class, it can do so even when objects are only observed from partial views. In a second experiment, we train our language model with a set of depth images taken only from the front of objects and can successfully predict attributes given test depth images taken from the rear. This view-invariance is a key property afforded by our use of an explicitly learned 3D representation—traditional fully supervised approaches are not capable of handling this scenario. Using our system, we demonstrate a Baxter robot successfully determining which object to pick based on a Microsoft Kinect depth image of several candidate objects and a simple natural language description of the desired object as shown in Figure 1. Our system is fast, with inference running in under 100ms for a single language+depth-image query.

II. Background and Related Work

Object retrieval refers to the task of finding and recovering an object specified by a human user, generally using natural language. While it is an important problem within artificial intelligence, it is particularly germane to robots, which have the ability to physically interact with specified objects.

The computer vision and natural language grounding communities attempt to solve object retrieval by locating or *grounding* the object specified in an image using natural language [1], [2]. Krishnamurthy and Kollar [1] use a dataset of segmented objects and their natural language descriptions to learn the grounding of words to objects in the image by exploiting repeated occurrences of segmented objects within images, along with their descriptions in natural language. Hu et al. [2] use a similar approach albeit using deep neural networks to avoid parsing and feature construction by hand. Chen et al. [5] learn joint embeddings of language descriptions and colored 3D objects. These approaches are trained on image data collected from internet sources and may differ from data observed on a robot's camera. Moreover, these approaches are either 2D, making them sensitive to viewpoint differences, or not applicable to partially observed objects, making them difficult to apply to a real robot. Another relevant line of work is the SUN scene attribute dataset [6] which maps images to attributes such as "hills", "houses", "bicycle racks", "sun", etc. Such understanding of image attributes provides high level scene descriptions, for example "human hiking in a rainy field". However, these approaches are typically unsuited for direct robotics application as they require explicit examples of similar objects from many viewpoints, all annotated with language, which is an expensive labeling task. Indeed, the typical robotics approach involves creating a database of all objects in the robot's environment and retrieving the object that fits the human requester's description [7], [8], [9], [10].

Bore et al. [7] use a mobile robot to create a scene with segmented 3D objects, where an object can be retrieved on demand using a query object, thus bypassing language grounding entirely. Whitney et al. [10] identifies objects in a scene using their descriptions given by a human user; the system uses Partially Observable Markov Decision Processes (POMDPs) to ask questions to disambiguate the correct object. The system itself does not perform any classification, instead it uses POMDPs to choose the correct object after it is confident of the human's intention. Such an interactive language grounding has been improved by approaches such as Shridhar and Hsu [8] and Hatori et. al [9]. They use a corpus of natural language question answering along with images of the objects in the robot's view to learn object grounding with questions and answers without a POMDP, by classifying the best fitting object and asking questions based on a handcrafted policy. The improvements here are that the grounding occurs in sensor data observed by the robot without any handcrafted features. However, the agents in these methods have seen all instances within the object set. These systems would have a hard time grasping or identifying a novel object given just a partial view, which is a common in robotics.

Separately from language grounding, recent years have seen an uptick of interest in the computer vision community on the problem of partial object completion. These methods take a partially observed object model (either in voxelized form or via a depth image) and attempt to predict an entire 3D model for the observed object. The most common of these approaches are *object-database* methods which construct an explicit database of complete object scans and treat novel objects as queries into the database [11], [12], [13]. While these approaches can be successful if the query is very similar to a previously encountered object, they tend to have issues if objects are sufficiently novel. More recently, work has progressed on learned models that predict 3D shape from partially-observed input [14], [15], [16], [3]. While most of these methods require known voxelization of the partially observed query object, recent work has allowed for prediction directly from a depth image, an important requirement for most robotic applications [17].

Bayesian Eigenobjects (BEOs) [3], [17] offer compact representations of objects. Using 3D object models for training, BEOs generate a low-dimensional subspace serving as a basis which well captures the object classes used to train it. This decomposition is achieved via Variational Bayesian Principle Component Analysis (VBPCA) [18], ensuring that the learned subspace does not over-fit the training data. BEOs enable a number of useful features for robot perception and manipulation tasks, namely accurate object classification, completion, and pose estimation from object depth-map data. This makes the BEO subspace a compelling target for language mapping in a robotics domain. Critically, the hybrid variant of BEOs (HBEOs) [17] learn an explicit subspace like BEOs, but use a deep convolutional network to learn an embedding directly from a depth image into the object-subspace, allowing for high performance, fast runtime, and the ability to complete objects without requiring voxelization of observed objects.

III. Methods

Our objective is to disambiguate between objects based on depth images and natural language descriptions. The naive approach would be to directly predict an object depth image, or its representation, given the object's natural language description. However, this approach is difficult to train, as depth-data is high-dimensional and highly continuous in nature. Even more critically, such an approach would require language+depth-image pairs with a large amount of viewpoint coverage, an unreasonable task given the difficulty of collecting rich human-annotated descriptions of objects. Instead, our approach factors the representation, allowing it to learn to reason about 3D structure from non-annotated 3D models, to learn a viewpoint-invariant representation, BEO, of object shape. We combine this representation with a small set of language data to enable object-language reasoning.

Given a natural language phrase and a segmented depth image, our system maps the depth image into a compact viewpoint-invariant object representation and then produces a joint embedding: both the phrase and the object representation are embedded into a shared low-dimensional space. Language descriptions and depth images that correspond will be close in this space, while pairs of depth images and descriptions that do not correspond will be relatively far apart.

We achieve this relationship by forcing the low dimensional representations of a given object's depth-data and its referring natural language data to align closely with each other by reducing a similarity metric during training. At inference time, we compute this similarity metric between a natural language command and candidate objects, observed via a depth image, in the joint low-dimensional embedding. We then select the object that has the most similar depth-data to the given natural language command within our learned embedding. Our approach also allows the prediction of object attributes based on a depth image of an object or a natural language description of that object.

The language grounding system (shown in Figure 2) consists of two primary portions: An object embedding portion (BEO module) which maps a partially observed object to a point in low-dimensional embedding space, and a language-BEO embedding portion which maps these embedded objects into attribute-based language descriptions. The two portions are trained in two different phases; first the BEO module is trained from a set of non-annotated 3D object models, and then the second phase learns a joint embedding of the natural language descriptions of the objects and their corresponding BEO representations. The training data used for these two phases can be completely disjoint; there is no requirement that all objects used to train the BEO module are also used to train the language mapping portion. This permits a larger corpus of 3D shape data to be used with a small set of human-annotated data. Because the BEO module, once trained, can produce predictions from a single depth image, the human annotations need not be over entire 3D objects, but could instead be assigned to individual images for which no true 3D model is known. These annotations need not cover the entire range of viewpoints a robot might encounter, and could be from only a very limited set of views, because the BEO module provides for generalization across viewpoints.

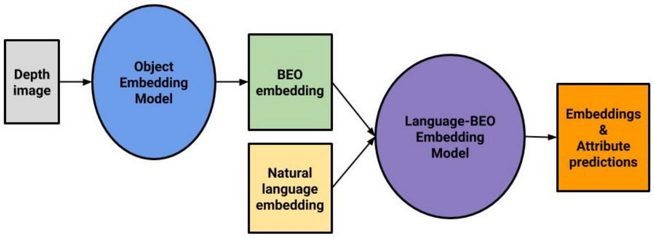
Depth image
Object Embedding Model
BEO embedding
Natural language embedding
Language-BEO Embedding Model
Embeddings & Attribute predictions

Fig. 2. Diagram of the language grounding pipeline, consisting of the object and language-BEO embedding models.

A. Obtaining Low-Dimensional BEO Embeddings

The low-dimensional BEO embedding is learned from a set of 3D meshes from a single class. These meshes are aligned and voxelized as in Burchfiel and Konidaris [17]. Each voxelized object is then stacked into a vector-representation; an object is now a point in high-dimensional voxel space. We perform VBPCA on these object points to yield a low-dimensional subspace defined by a mean vector, $\boldsymbol{\mu}$ and basis matrix, $\mathbf{W}$. We find an orthnormal basis $\mathbf{W}' = orth([\mathbf{W}, \boldsymbol{\mu}])$ using singular value decomposition and, with slight abuse of notation, hereafter refer to $\mathbf{W}'$ as simply $\mathbf{W}$. Given a new (fully observed) object $\mathbf{o}$, we can obtain its embedding $\mathbf{o}'$ in this space via

$$\mathbf{o}' = \mathbf{W}^T \mathbf{o}, \tag{1}$$

and any point in this space can be *back-projected* to 3D voxel space via

$$\hat{\mathbf{o}} = \mathbf{W}\mathbf{o}'. \tag{2}$$

We use the HBEONet network structure [17] to then learn a deep model for partially-observable inference which, given a depth image, directly predicts an embedding into the object subspace. During training, HBEONet receives depth images (generated synthetically from the 3D models) and target object embeddings. It consists of several convolutional and

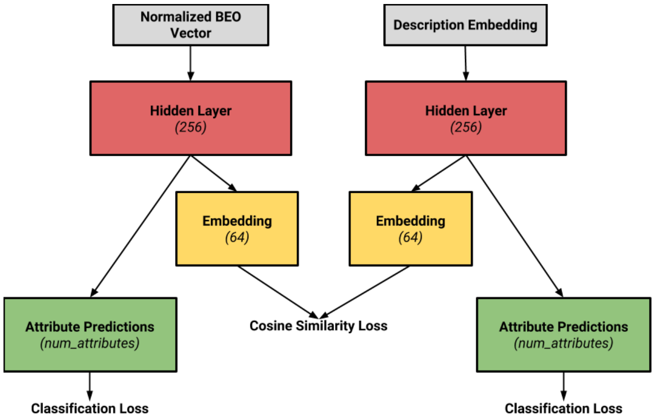

Fig. 3. Diagram of the language-BEO embedding model. After training, the model outputs the cosine-similarity between a given language description and the BEO embedding of an object. The model can also predict attributes of an object given its BEO vector, or the object attributes given the natural language used to refer to the object. *num_attributes* is number of attributes to be predicted.

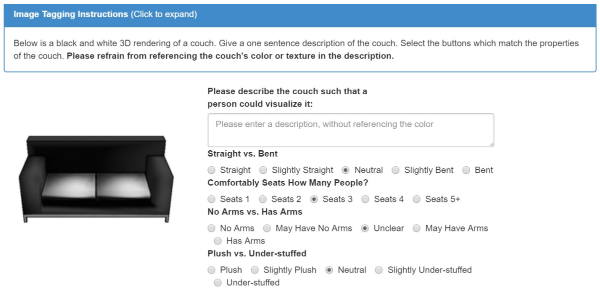

Image Tagging Instructions (Click to expand)

Below is a black and white 3D rendering of a couch. Give a one sentence description of the couch. Select the buttons which match the properties of the couch. **Please refrain from referencing the couch's color or texture in the description.**

Please describe the couch such that a person could visualize it:

Please enter a description, without referencing the color

Straight vs. Bent

○ Straight ○ Slightly Straight ● Neutral ○ Slightly Bent ○ Bent

Comfortably Seats How Many People?

○ Seats 1 ○ Seats 2 ● Seats 3 ○ Seats 4 ○ Seats 5+

No Arms vs. Has Arms

○ No Arms ○ May Have No Arms ● Unclear ○ May Have Arms ○ Has Arms

Plush vs. Under-stuffed

○ Plush ○ Slightly Plush ● Neutral ○ Slightly Under-stuffed ○ Under-stuffed

Fig. 4. Amazon Mechanical Turk data collection interface. Workers were shown full 360-degree views of the 3D objects.

fully connected layers [17]. This network is then used to train the language grounding portion of the system.

B. Combining BEO Representations and Language

The BEO representations received from HBEONet are low-dimensional representations of object 3D meshes. Similarly, language can be represented in low-dimensional embeddings using techniques like Word2Vec [19], [20], or GloVe [21]. However, the BEO embedding of an object has no connection to or similarity with the language embedding of the same object's natural language description. The BEO embeddings for a class of objects might be on a completely different manifold than the language embeddings for the descriptions of those objects, even if the dimensionality of the embedding spaces is the same. This discrepancy exists because these manifolds are learned for different tasks: BEO embeddings are learned for compactness of 3D object representation, whereas the language embeddings are learned for the predictive power of word representations in the context of a language model.

To overcome this and learn a joint embedding space, we take an approach inspired by Siamese neural networks [22]. We train a joint model which learns new embeddings for both BEO vectors and natural language. Embeddings of language descriptions and BEO vectors for similar objects should be close together in this new embedding space, whereas embeddings of BEO vectors and language descriptions that do not correspond to similar objects should be farther apart. To measure closeness in this space, we use *cosine similarity*:

$$\textit{cosine\_similarity}(\boldsymbol{x},\boldsymbol{y}) = \frac{\boldsymbol{x}\cdot\boldsymbol{y}}{|\boldsymbol{x}|\cdot|\boldsymbol{y}|} \tag{3}$$

The cosine similarity of two vectors is defined as the cosine of the angle between the vectors. It is calculated as the inner product of the vectors, normalized by the product of their lengths. The two vectors are maximally similar when their cosine similarity is 1, and maximally dissimilar when their cosine similarity is -1.

Given an object's depth image and its corresponding language description, we first compute the low-dimensional BEO vector representation for the depth-data and then pass this representation to our joint embedding model shown in Figure 3. This model takes as input a BEO vector and a bag-of-words sentence embedding of the natural language description. For sentence embeddings we use the mean GloVe [21] embedding of all the words in a sentence. GloVe embeddings have been shown to be a robust sentence level feature [23]. Both the BEO vector and the sentence embedding are passed through hidden layers of size 256, and the embedding dimension is then further reduced to size 64. The cosine similarity loss is calculated between the 64 dimensional embeddings of language and BEO vectors. We also predict object attributes directly from the 256 dimensional hidden layers for language and BEO vectors (separately), which we train via binary cross-entropy loss. As a result, for each of the inputs (natural language description and depth image), two outputs are produced: a 64 dimensional joint embedding, and a prediction over the six attributes selected for each object class.[1]

C. Data Collection and ShapeNet Object Meshes

We used three classes of objects from ShapeNet as our primary dataset: couches, cars, and airplanes. These models consist of 3D meshes, aligned and scaled to relatively consistent size. ShapeNet is a fairly noisy dataset with significant amounts of variation; it is not unusual to find models with extremely unusual shapes and some models may be incomplete. We manually removed objects from each category that were incorrectly classified or had significant damage to their meshes, but retained odd and unusual examples. During BEO training, each object was voxelized to a resolution of $64 \times 64 \times 64$ and 300 dimensional object-subspaces were used, capturing between 80% — 90% of training object variance.

We collected natural language text descriptions and attribute labels, for the purpose of discriminating between dif-

[1] Each input modality, language and depth, takes an independent path through the network. Therefore, we can obtain our 64-dimensional embeddings and attribute predictions from just a BEO vector or language vector, which is required during attribute classification.

TABLE I

OBJECT RETRIEVAL RESULTS

Object Class	Human Baseline	Full-view		Partial-view		View-transfer	
	Top-1 (Std. Error)	***Top-1***	***Top-2***	***Top-1***	***Top-2***	***Top-1***	***Top-2***
Couch	77.3% (0.025)	60.6%	87.1%	58.6%	86.3%	57.6%	85.1%
Car	76.0% (0.024)	74.6%	93.8%	73.5%	93.4%	72.2%	93.1%
Airplane	69.3% (0.027)	66.7%	92.8%	67.0%	92.5%	68.3%	92.7%

ferent objects using language descriptions, through Amazon Mechanical Turk (AMT). The data collection interface can be seen in Figure 4. AMT workers were shown full 360° views of the 3D object instances, and got to select from a 5-point scale for each attribute classification. The attribute labels were collected to serve two purposes: First, they allow us to observe the amount of object attribute signal present in the BEO representation. Second, we wanted to guide the natural language descriptions provided by the AMT workers, as otherwise the workers are likely to just describe an object with its name and not provide any object discriminative language. The attributes were chosen such that they describe physical and abstract properties of an object. For example, physical attributes such as long/short and abstract attributes such as luxury/economical were chosen for cars. We selected attributes which had the highest inter-rater agreement, as we wanted the attribute selections to have the highest consistency among English speakers. The chosen object attributes are listed in Table II. AMT data were collected for 1250 couch, 3405 car, and 4044 plane object instances. We collected 10 sentence and attribute descriptions for each object instance.

IV. EXPERIMENTS AND RESULTS

Our evaluation was designed to verify that our combined model can accurately select an object given a natural language description of the object and depth image observations of possible candidate objects. We evaluate our model on two tasks: object retrieval and attribute classification. During object retrieval, the system observes three possible objects and must select the object corresponding to an input natural-language description. When performing attribute classification, the system observes a single object and must correctly predict its attributes.

We split our collected AMT data using 70% for training, 15% for development, and 15% for testing. Training examples for the language portion of the model consisted of pairs of depth-image-induced BEO object vectors and corresponding language descriptions, along with the mean of the crowd-sourced attributes, binarized to signify the presence or absence of an attribute. Negative examples were generated by randomly selecting depth images and language descriptions that do not correspond. An equal number of positive and negative examples were shown to the network over the course of training. The cosine similarity loss we employed ensures that positive examples have higher cosine similarities and negative examples have lower cosine similarities in the 64 dimensional space.[2] We then trained the language-object embedding of our model using 5 randomly sampled depth images—and their resulting BEO shape vectors—from each object in the training set, along with the natural language description, and the binarized mean of human-provided attribute labels, corresponding to each depth image.

For the retrieval task, we show results for three training conditions: 1) *Full-view*: a baseline where the system is given a ground-truth 3D model for the object it is observing, 2) *Partial-view*: the scenario where the system is trained and evaluated with synthetic depth images over a wide variety of possible viewpoints for each object, 3) *View-transfer*: the system is identical to the previous *partial-view* case, except all training images come from a frontal viewpoint and all evaluation images are obtained from a side-rear view. We train our model separately on all three object classes we sourced from ShapeNet. In all experiments, the input BEO embeddings to the language portion of our network were trained directly from all non-language-annotated meshes in the training dataset. In the partial-view and view-transfer cases, HBEONet was then trained using 600 synthetically rendered depth images, across a variety of viewpoints, from each 3D mesh. The baseline, full-view case, uses equation 1 instead of HBEONet to transform object observations into BEO embeddings.

To demonstrate the ability of our system to produce reasonable language attributes given a single query depth image, we also evaluate our model's attribute prediction performance against a human baseline.

Finally, to show the applicability of our approach to a real robot, we tested our system on a Baxter robot and demonstrated successful discrimination between three candidate objects given natural language descriptions and a Microsoft Kinect-generated depth image.

A. Object Retrieval from Natural Language

To evaluate the retrieval performance of our model, we randomly selected a set of 10 depth images for each object in our test set. In the full-view case, we simply used the 3D model for each object in the test set.

In each retrieval test, we showed the system three different objects along with a natural language description of one of those objects. We report the accuracy of our model in selecting the correct object in Table I for both top-1 and top-2 retrieval metrics.

[2] Training employed the Adam optimizer, with a learning rate of 0.0001, and proceeded until accuracy on the development set began to decrease.

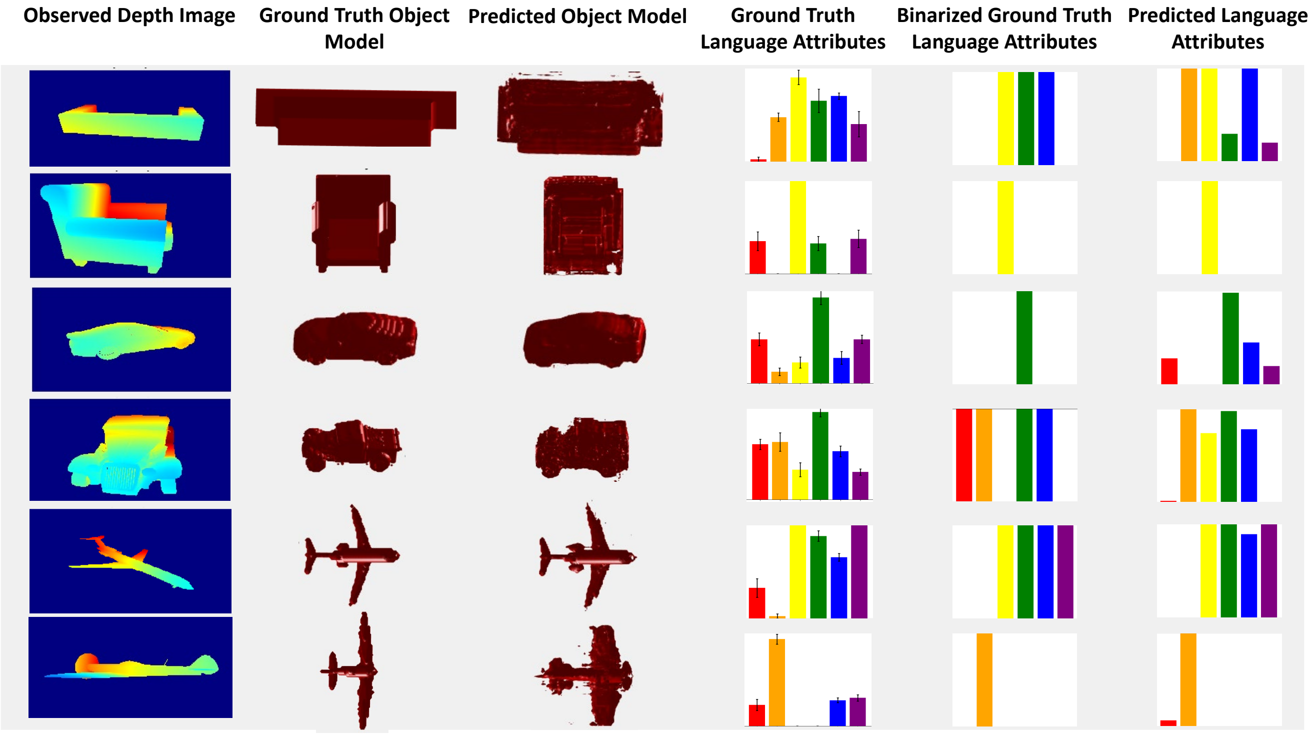

Fig. 5. Non-cherry-picked attribute prediction examples, produced by our model from single depth image input. The Predicted Object Model column visualizes BEO-predicted embeddings projected back into voxel space via equation 2.

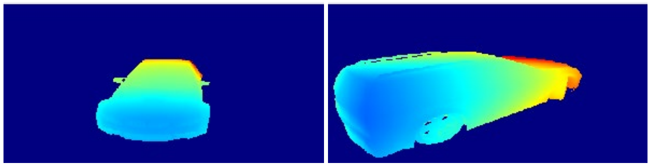

Fig. 6. Example training depth image (left) and test depth image (right) from the view-transfer experiment. The underlying BEO representation maps these substantially different viewpoints to the same embedding space, helping the language grounding network ground object names and attributes to objects from unseen viewpoints.

We also evaluated the robustness of our system to substantial viewpoint differences between testing and training data by training our language model with only frontal views while evaluating model performance based only on rear-side views. Figure 6 shows example training and testing depth images from our view-transfer experiment for a car object instance. Because the BEO representation is viewpoint invariant, our system successfully performs this task despite the dramatic distribution change between the training and evaluation datasets; see View-transfer results in Table I.

We found that not only did the performance of our system decline only slightly between the fully-observable-object (full-view) baseline and the depth image-based partial-view scenario, training our model with annotated images from a single viewpoint (view-transfer) also failed to dramatically reduce performance. These results constitute a strong indicator that our system is successfully achieving view invariance and generalization by relying on the larger set of unlabeled object meshes used to train the BEO embeddings.

B. Human Retrieval Baseline

We compare the performance of our system to a human baseline for the retrieval task. Humans are expert symbol grounders and are able to ground objects from incomplete descriptions rather well from an early age [24]. We showed human users (full 360-degree views of) three objects and one language description, where the language was collected from AMT for one of the objects shown, and asked them to pick the object to which the language refers. The sample size for this experiment is 300 users, and the study was done on AMT. The results are shown in Table I. We found human performance similar to our system's top-1 retrieval accuracy in the full-view case. Even human users are not perfect at this task, as ambiguity exists between objects and their descriptions in many retrieval scenarios; because the set of three objects to be retrieved are chosen at random, there are often multiple very similar objects that must be disambiguated, resulting in multiple reasonable predictions.

C. Object Attribute Prediction

We also evaluated the 6-category object attribute predictions produced by our system; we predict language attributes for each of our object classes; Table II summarizes our system's performance on this task, while Figure 5 illustrates several example attribute predictions produced by our system given a single query depth image. Attributes in each

TABLE II

Chosen Object Attributes and Our System's Respective Attribute Prediction F1 Scores

Object Class	Attr. 1	Attr. 2	Attr. 3	Attr. 4	Attr. 5	Attr. 6
Couch	Straight/Bent **0.87**	Seats 1-5+ **0.87**	No-Arms/Has-Arms **0.80**	Plush/Understuffed **0.64**	Short/Long **0.83**	Simple/Ornate **0.67**
Car	Long/Short **0.81**	Curvy/Boxy **0.83**	Fast/Slow **0.83**	Emergency/Not **0.91**	Luxury/Economy **0.77**	Seats 1-5+ **0.83**
Airplane	Rounded/Angular **0.88**	Commercial/Military **0.91**	Propeller/Jet **0.94**	Historical/Modern **0.93**	Engines 0-4+ **0.73**	Seats 0-4+ **0.88**

histogram are in the order, from left to right, that they are listed in Table II. The first column of Figure 5 illustrates the input depth image given to the system, the second column the observed object's ground truth shape, the third column the shape back-projected, via Equation 2 from the predicted BEO embedding, the fourth column contains raw attribute labels gathered from human-labelers for that object, the fifth column show these human-annotate labels after being binarized, and the sixth column shows the attribute predictions output by our system. Note that the system is trained with binarized attribute labels. The system was generally quite accurate when performing attribute predictions; it tends to be most confident in scenarios where human-labelers were highly mutually consistent, which likely resulted in increased consistency among attribute labels for similar objects in the training data.

D. Picking Physical Objects from Depth Observations and Natural Language Descriptions

We implemented our system on a Baxter robot: a mechanically compliant robot equipped with two parallel grippers. For this evaluation, we obtained realistic model couches (designed for use in doll houses) to serve as our test objects. We use the same (synthetically trained) network employed in the prior experiments, without retraining it explicitly on Kinect-generated depth images. We passed a textual language description of the requested object into our model along with a manually-segmented and scaled Kinect-captured depth image of each object in the scene. The robot then selected the observed object with the highest cosine-similarity with the language description and performed a pick action on it. Our system successfully picked up desired objects using phrases such as "Pick up the couch with no arms." Despite having no doll-house couches or Kinect-produced depth images in its training data, our system was able to generalize to this new domain.

V. Future Directions

One of the critical challenges in robotics is scaling techniques to work in a wide array of environments. While the results presented in this paper constitute an important step towards grounding objects to natural language, existing work in this area still generally only applies to a small number, 1 — 4, of object classes [5], [25]. Moving forward, more expansive datasets, of both natural language object descriptions and 3D shapes, are necessary to enable larger systems to be learned. While the ShapeNet object database is incredibly useful, it is still small by modern machine learning standards, containing roughly 50 classes (only 10 of which have at least 1k examples) and 50k object models. Just as the introduction of ImageNet, with a thousand classes and a million images, pushed the state of the art in 2D classification forward, a similarly sized shape dataset would be invaluable for 3D object understanding.

VI. Conclusion

We developed a system to ground natural language descriptions of objects into a common space with observed object depth images. This scenario is challenging because novel objects will only be observed from a single, possibly unusual, viewpoint. Our approach decouples 3D shape understanding from language grounding by learning a generative 3D representation (using BEOs) that maps depth images into a viewpoint-invariant low-dimensional embedding. This 3D representation is trainable from non-annotated 3D object meshes, allowing us to label only a small subset of object depth images compared to the whole dataset. This viewpoint invariance and data efficiency is key due to the difficulty of acquiring large volumes of human-annotated labelling.

We show that our system successfully discriminates between objects given several segmented candidate depth images and a natural language description of a desired item, performing almost as well as a human baseline. This object discrimination is possible even in challenging scenarios where the training depth images and evaluation depth images are selected from entirely different viewpoints. We also demonstrate our method on a Baxter robot and show that the robot is able to successfully pick the correct item repeatedly based on a natural language description.

Acknowledgments

We would like to acknowledge Edward Williams for his role in the conception of this project. This research was supported by the National Aeronautics and Space Administration under grant number NNX16AR61G, and in part by DARPA under agreement number D15AP00104. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The content is solely the responsibility of the authors and does not necessarily represent the official views of DARPA.

References

[1] J. Krishnamurthy and T. Kollar, "Jointly learning to parse and perceive: Connecting natural language to the physical world," *Transactions of the Association for Computational Linguistics*, vol. 1, pp. 193–206, 2013.

[2] R. Hu, H. Xu, M. Rohrbach, J. Feng, K. Saenko, and T. Darrell, "Natural language object retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4555–4564.

[3] B. Burchfiel and G. Konidaris, "Bayesian eigenobjects: A unified framework for 3D robot perception," in *Proceedings of Robotics: Science and Systems*, 2017.

[4] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "Shapenet: An information-rich 3D model repository," *arXiv:1512.03012*, 2015.

[5] K. Chen, C. B. Choy, M. Savva, A. X. Chang, T. Funkhouser, and S. Savarese, "Text2shape: Generating shapes from natural language by learning joint embeddings," *arXiv:1803.08495*, 2018.

[6] G. Patterson and J. Hays, "Sun attribute database: Discovering, annotating, and recognizing scene attributes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2751–2758.

[7] N. Bore, R. Ambrus, P. Jensfelt, and J. Folkesson, "Efficient retrieval of arbitrary objects from long-term robot observations," *Robotics and Autonomous Systems*, vol. 91, pp. 139 – 150, 2017.

[8] M. Shridhar and D. Hsu, "Interactive visual grounding of referring expressions for human-robot interaction," *arXiv:1806.03831*, 2018.

[9] J. Hatori, Y. Kikuchi, S. Kobayashi, K. Takahashi, Y. Tsuboi, Y. Unno, W. Ko, and J. Tan, "Interactively picking real-world objects with unconstrained spoken language instructions," in *Proceedings of International Conference on Robotics and Automation*, 2018.

[10] D. Whitney, E. Rosen, J. MacGlashan, L. L. Wong, and S. Tellex, "Reducing errors in object-fetching interactions through social feedback," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2017, pp. 1006–1013.

[11] Q. H. Y. Kim, N. J. Mitra and L. Guibas, "Guided real-time scanning of indoor objects," in *Computer Graphics Forum*, vol. 32, 2013, pp. 177–186.

[12] D. M. Y. Y. Kim, N. J. Mitra and L. Guibas, "Acquiring 3D indoor environments with variability and repetition," *ACM Transactions on Graphics*, vol. 31, pp. 138:1–138:11, 2012.

[13] Y. Li, A. Dai, L. Guibas, and M. Nießner, "Database-assisted object retrieval for real-time 3D reconstruction," in *Computer Graphics Forum*, vol. 34, 2015, pp. 435–446.

[14] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3D shapenets: A deep representation for volumetric shapes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1912–1920.

[15] A. Dai, C. Qi, and M. Nießner, "Shape completion using 3D-encoder-predictor CNNs and shape synthesis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[16] A. Soltani, H. Huang, J. Wu, T. Kulkarni, and J. Tenenbaum, "Synthesizing 3D shapes via modeling multi-view depth maps and silhouettes with deep generative networks," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2511–2519, 2017.

[17] B. Burchfiel and G. Konidaris, "Hybrid bayesian eigenobjects: Combining linear subspace and deep network methods for 3D robot vision," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2018, pp. 6843–6850.

[18] C. M. Bishop, "Variational principal components," in *Proceedings of the International Conference on Artificial Neural Networks*, 1999, pp. 509–514.

[19] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv:1301.3781*, 2013.

[20] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proceedings of the International Conference on Advances in Neural Information Processing Systems*, 2013, pp. 3111–3119.

[21] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2014, pp. 1532–1543.

[22] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a siamese time delay neural network," in *Proceedings of the International Conference on Advances in Neural Information Processing Systems*, 1994, pp. 737–744.

[23] J. Wieting, M. Bansal, K. Gimpel, and K. Livescu, "Towards universal paraphrastic sentence embeddings," *arXiv:1511.08198*, 2015.

[24] H. Rakoczy, M. Tomasello, and T. Striano, "How children turn objects into symbols: A cultural learning account," *Emory Symposia in Cognition. Symbol Use and Symbolic Representation: Developmental and Comparative Perspectives.*, pp. 67–97, 2005.

[25] M. E. Yumer, S. Chaudhuri, J. K. Hodgins, and L. B. Kara, "Semantic shape editing using deformation handles," *ACM Transactions on Graphics*, vol. 34, no. 4, p. 86, 2015.