ZSON: Zero-Shot Object-Goal Navigation using Multimodal Goal Embeddings

Anonymous Author(s)

Abstract: We present a scalable approach for learning open-world object-goal navigation (ObjectNav) – the task of asking a virtual robot (agent) to find any instance of an object in an unexplored environment (e.g., “find a sink”). Our approach is entirely zero-shot – i.e., it does not require ObjectNav rewards or demonstrations of any kind. Instead, we train on the image-goal navigation (ImageNav) task, in which agents find the location where a picture (i.e., goal image) was captured. Specifically, we encode goal images into a multimodal, semantic embedding space to enable training semantic-goal navigation (SemanticNav) agents at scale in unannotated 3D environments (e.g., HM3D). After training, SemanticNav agents can be instructed to find objects described in free-form natural language (e.g., “sink,” “bathroom sink,” etc.) by projecting language goals into the same multimodal, semantic embedding space. As a result, our approach enables open-world ObjectNav. We extensively evaluate our agents on three ObjectNav datasets (Gibson, HM3D, and MP3D) and observe absolute improvements in success of 4.2% - 20.0% over existing zero-shot methods.

1 Introduction

Imagine asking a home assistant robot to find a “flat-head screwdriver” or the “medicine case near the bathroom sink.” Building such assistive agents is a problem of deep scientific and societal value.

To study this problem systematically, the embodied AI community has rallied around a problem called object-goal navigation (ObjectNav) [1]. Given the name of an object (e.g., “chair”), ObjectNav involves exploring a 3D environment to find any instance of the object. The last few years have witnessed the development of new environments [2, 3, 4, 5, 6], annotated 3D scans [7, 8, 9], datasets of human demonstrations [10], and approaches for ObjectNav [11, 12, 13, 14, 15, 16], cumulatively leading to strong progress. For instance, the entries in the annual Habitat challenge [17] have jumped from 6% success (DD-PPO baseline in 2020) to 53% success (in ongoing 2022 Habitat Challenge).

While this progress is exciting, we believe that a subtle but insidious assumption has snuck into this line of work: the closed-world assumption. We started by discussing an open-world scenario where a person may describe any object in language (e.g., “flat-head screwdriver”), but ObjectNav is currently formulated over a closed predetermined vocabulary of object categories (“chair”, “bed”, “sofa”, etc.), with approaches using pre-trained object detectors and segmenters for these categories [10, 11, 12, 13]. While this assumption may have been essential to get started on this problem, it is now important to move beyond it and ask – how can embodied agents find objects in an open-world setting?

In this work, we develop an approach for ObjectNav that is both zero-shot, i.e., does not require any ObjectNav rewards or demonstrations, and open-world, i.e., does not require committing to a taxonomy of categories. Our key insight is that we can create a visiolinguistic embedding space to decouple two problems – (1) describing and representing semantic goals (“chair”, “brown chair”, picture of brown chair) from (2) learning to navigate to semantic goals.

Submitted to the 6th Conference on Robot Learning (CoRL 2022). Do not distribute.
To represent semantic goals (1), we leverage recent advances in multimodal AI research on learning a common embedding space for images and text using large collections of image-captions pairs. Specifically, we use CLIP [18], a method for training dual vision and language encoders that produce similar representations for paired data such as an image and its caption. We use CLIP to transform image-goals (e.g., a picture of the kitchen island) and object-goals (e.g., "bathroom sink") into semantic-goals representing navigation targets. Our main observation is that a semantic-goal produced from an image (e.g., a picture of the bathroom sink) should be similar to semantic goals produced from descriptions of the same target (e.g., "bathroom sink"). Thus, we hypothesize that these modalities (images and language) can be used interchangeably for creating semantic goals.

Accordingly, for learning to navigate to semantic goals (2), we train agents using image-goals encoded via CLIP’s image encoder. Then, we evaluate the learned navigation policy on ObjectNav, where goals are specified in language (e.g., “chair”) and encoded via CLIP’s text encoder. As a result, our agents perform ObjectNav without ever directly training for the task – i.e., in a zero-shot manner.

We perform large-scale experiments on three ObjectNav datasets – Gibson [4], MP3D [8], and HM3D [19]. Our zero-shot agent (that has not seen a single 3D semantic annotation or ObjectNav training episode) achieves a 31.3% success in Gibson environments, which is a 20.0% absolute improvement over previous zero-shot results [20]. In MP3D, our agent achieves 15.3% success, a 4.2% absolute gain over existing zero-shot methods [21]. For reference, these gains are on par or better than the 5% improvement in success between the Habitat 2020 and 2021 ObjectNav challenge winners. On HM3D, our agent’s zero-shot SPL matches a state-of-the-art ObjectNav method [16] that trains with direct supervision from 40k human demonstrations.

2 Related Work

Zero-Shot ObjectNav. Two recent works [20, 21] directly address our motivation (zero-shot ObjectNav) and are most related. First, ZER [20] proposes a two-stage framework in which an image-goal navigation (ImageNav) agent is first trained from scratch. Then, independent encoders are trained to map from various modalities (including language) into the image-goal embedding space. A key challenge with this approach is that image-goal embeddings may not capture semantic information because semantic annotations are not used in ImageNav training. Instead, an ImageNav agent trained from scratch may learn to pattern match visual observations and goal image embeddings. By contrast, our approach reverses these two stages, with CLIP pretraining representing stage one. Thus, our approach uses a goal embedding space that captures semantics by design. We empirically demonstrate the benefits of our proposed approach in Section 4.

In concurrent work, CLIP-on-Wheels (CoW) [21] uses a gradient-based visualization technique (GradCAM [22]) with CLIP to localize objects in the agent’s observations. This is combined with a
heuristic exploration policy to enable zero-shot object-goal navigation. In contrast, we demonstrate that learning a navigation policy can substantially outperform the heuristic exploration approach proposed in [21] without using explicit object localization techniques.

3 Approach

This section describes our framework for training visual navigation agents. We use CLIP [18] to produce semantic goal embeddings of image-goals (e.g., a picture of the sink) and object-goals (e.g., “sink”). This allows training semantic-goal navigation agents at scale using image-goals in HM3D environments [19], then deploying these agents for object-goal navigation in a zero-shot manner. In other words, our agents execute object-goal navigation without ever directly training for the task.

Learning Semantic-Goal Navigation As illustrated in Fig. 1 (top-left), given an image-goal $v^G$, we use a CLIP visual encoder $\text{CLIP}_v$ to generate a semantic goal embedding $s^G_t = \text{CLIP}_v(v^G)$ that is used to guide navigation. Conceptually, encoding image-goals with CLIP preserves semantic information about the goal, such as visual concepts that might be described in image captions (e.g., “a sofa in a living room”). However, semantic goal embeddings are less likely to include low-level features (e.g., the exact patterns in a wood floor) that do not correlate with web-scraped captions.

While removing low-level information might make the pretraining task more difficult, our goal is to learn a policy that transfers to ObjectNav in which agents only receive high-level goals (e.g., “Find a sofa”). As an added benefit, generating semantic goal embeddings as a pre-processing step substantially improves training time (by $\sim 3.5x$).

Our agent architecture is shown in Fig. 1. At each timestep $t$, our agent receives an egocentric RGB observation $v_t$ and a goal representation $s^G_t$. The observation is processed by a ResNet-50 [23] encoder, which is pretrained on the Omnidata Starter Dataset (OSD) [24] using self-supervised learning (DINO [25]) following the pretraining recipe presented in OVRl [16]. The output from the ResNet-50 encoder is concatenated with the goal representation $s^G_t$ and an embedding of the agent’s previous action $a_{t-1}$ and then passed to the policy network composed of a two-layer LSTM. The policy network outputs a distribution over the action space. We train our SemanticNav agent with reinforcement learning (RL). Specifically, we train with DD-PPO [26] using two data augmentation techniques: color jitter and random translation (adapted from [16]).

Zero-Shot Object-Goal Navigation In ObjectNav [1], agents are given a target category (e.g., “sofa” or “chair”) and must locate any instance of that object (i.e., “any sofa” or “any chair”). Similar to ImageNav, ObjectNav requires exploring new environments that the agent has never seen before. However, in ObjectNav, the goal (e.g., “sofa”) provides a minimal amount of information about where the agent must go and it requires recognizing any version of the goal object in the new scene.

To address this task, we transform object-goals $o^G$ (e.g., “sofa”) into semantic goal embeddings using the CLIP text encoder $\text{CLIP}_t$, which results in the semantic goal $s^G_t = \text{CLIP}_t(o^G)$. CLIP aligns image and text, thus the semantic goals from text $s^G_t$ should be close (in terms of cosine similarity) to the CLIP visual embeddings $s^v_t$ used in training. To keep our approach simple and easily reproducible, we do not use any prompt engineering (e.g., using a template such as “A photo of a <>”). Instead, we simply use the object name (e.g., “sofa”) as the object-goal input $o^G$.

4 Experiments

Experimental Setup We training our SemanticNav agents using the 800 training environments from HM3D [19], and measure performance on one ImageNav and three ObjectNav datasets. This requires using two different agent embodiments termed configuration A and B below. We compare with, to the best of our knowledge, the only two existing zero-shot methods for object-goal navigation (ObjectNav): (1) Zero Experience Required (ZER) [20] and (2) CLIP on Wheels (CoW) [21].
Table 1: **Zero-shot ObjectNav performance** on Gibson [4], HM3D [19], and MP3D [8] validation. Our approach (ZSON) substantially improves on previous zero-shot methods and narrows the gap to SOTA fully-supervised methods such as OVRL [16], which is provided for reference.

<table>
<thead>
<tr>
<th>Method</th>
<th>ImageNav (Gibson) SPL</th>
<th>SR</th>
<th>ObjectNav (Gibson) SPL</th>
<th>SR</th>
</tr>
</thead>
<tbody>
<tr>
<td>OVRL [16]</td>
<td>27.0%</td>
<td>54.2%</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>ZER [20]</td>
<td>21.6%</td>
<td>29.2%</td>
<td>-</td>
<td>11.3%</td>
</tr>
<tr>
<td>ZSON (ours)</td>
<td><strong>28.0%</strong></td>
<td><strong>36.9%</strong></td>
<td><strong>12.0%</strong></td>
<td><strong>31.3%</strong></td>
</tr>
</tbody>
</table>

(a) Configuration A

<table>
<thead>
<tr>
<th>Method</th>
<th>ObjectNav (HM3D) SPL</th>
<th>SR</th>
<th>ObjectNav (MP3D) SPL</th>
<th>SR</th>
</tr>
</thead>
<tbody>
<tr>
<td>OVRL [16]</td>
<td>12.3%</td>
<td>32.8%</td>
<td>7.0%</td>
<td>25.3%</td>
</tr>
<tr>
<td>CoW [21] (w/depth)</td>
<td>-</td>
<td>-</td>
<td>6.3%</td>
<td>11.1%</td>
</tr>
<tr>
<td>ZSON (ours)</td>
<td>12.6%</td>
<td>25.5%</td>
<td>4.8%</td>
<td>15.3%</td>
</tr>
</tbody>
</table>

(b) Configuration B

**Zero-Shot Object-Goal Navigation Results** In Table 1a, we compare with ZER [20] using configuration A. Notice that our agent is stronger on ImageNav, the base pretraining task before ObjectNav can be studied. Specifically, we observe a 7.7% improvement in success rate SR (29.2% → 36.9%). This improvement results from (1) learning to navigate to semantic goal embeddings (as proposed in this work) instead of navigating to image-goal embeddings that are learned from scratch (as done in ZER), (2) using more diverse training environments, and (3) from using a pretrained visual encoder. We ablate factors (2) and (3) in the next, and observe improved performance from factor (1) alone. In Table 1a, we see even larger improvements in ObjectNav SR of 20.0% (11.3% → 31.3%). These results indicate that our design decisions are particularly useful for zero-shot ObjectNav.

In Table 1b we compare with CoW [21] using configuration B. On MP3D, we observe that ZSON improves ObjectNav SR by 4.2% absolute and 37.8% relative (11.1% → 15.3%). These results demonstrate that learning a navigation policy improves zero-shot ObjectNav SR over the hand-designed exploration strategy proposed by CoW. Moreover, we expect further improvements in zero-shot ObjectNav performance from scaling our approach (e.g., by collecting more training environments). On HM3D we find that our agent achieves a strong SR of 25.5% and SPL of 12.6%. Impressively, this zero-shot SPL matches OVRL [16], which is directly trained on 40k human demonstrations [10] for the ObjectNav task with imitation learning.

Table 2: **Comparison with ZER [20]** using a ResNet-9 and the Gibson dataset with our approach. Learning SemanticNav (Ours) outperforms learning ImageNav then language grounding (ZER [20]).

<table>
<thead>
<tr>
<th>Method</th>
<th>Visual Encoder</th>
<th>Training Dataset</th>
<th>ImageNav (Gibson) SPL</th>
<th>SR</th>
<th>ObjectNav (Gibson) SPL</th>
<th>SR</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZER [20]</td>
<td>ResNet-9</td>
<td>Gibson</td>
<td>21.6%</td>
<td>29.2%</td>
<td>-</td>
<td>11.3%</td>
</tr>
<tr>
<td>Ours</td>
<td>ResNet-9</td>
<td>Gibson</td>
<td><strong>22.8%</strong></td>
<td><strong>33.3%</strong></td>
<td><strong>7.4%</strong></td>
<td><strong>15.3%</strong></td>
</tr>
</tbody>
</table>

**Comparison with ZER without encoder pretraining or diverse training environments.** In Table 2, we train in Gibson environments (instead of HM3D) and do not use a pretrained observation encoder. These settings match ZER [20], allowing for a direct comparison between the two methods. We observe that our approach results in a 4.0% absolute and 35% relative improvement in zero-shot ObjectNav success (11.3% → 15.3%). These results demonstrate that learning to navigate to semantic-goal embeddings outperforms the inverse approach proposed by ZER of first training for ImageNav, then learning a mapping from object categories into the image-goal embedding space.

**Discussion.** We present a zero-shot method for learning open-world object-goal navigation (ObjectNav). Our approach involves projecting image-goals into a semantic-goal embedding space using an image-and-text alignment model (CLIP). This creates a semantic-goal navigation task that does not require annotated 3D environments or collecting human demonstrations. Thus, our method is easy to use for large-scale pretraining of visual navigation agents.
References


