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

Anonymous Author(s)
Affiliation
Address
`email`

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.

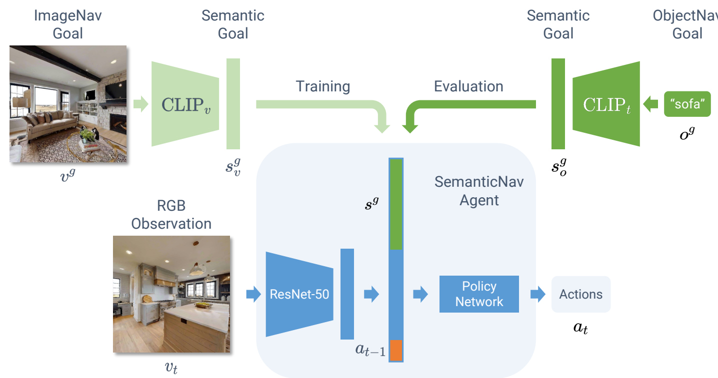

Figure 1: We tackle both `ImageNav` and `ObjectNav` via a common `SemanticNav` agent. This agent accepts a semantic goal embedding (s^g), which comes from either CLIP's visual encoder ($\texttt{CLIP}_v$) in ImageNav or CLIP's textual encoder ($\texttt{CLIP}_t$) in ObjectNav.

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 $\texttt{CLIP}_v$ to generate a semantic goal embedding $s_v^g = \texttt{CLIP}_v(v^g)$ 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 receives 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_v^g. 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_v^g 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 $\texttt{CLIP}_t$, which results in the semantic goal $s_o^g = \texttt{CLIP}_t(o^g)$. CLIP aligns image and text, thus the semantic goals from text s_o^g should be close (in terms of cosine similarity) to the CLIP visual embeddings s_v^g 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.

Method	ImageNav (Gibson)		ObjectNav (Gibson)	
	SPL	SR	SPL	SR
OVRL [16]	27.0%	54.2%	-	-
ZER [20]	21.6%	29.2%	-	11.3%
ZSON (ours)	**28.0%**	**36.9%**	12.0%	**31.3%**

(a) Configuration `A`

Method	ObjectNav (HM3D)		ObjectNav (MP3D)	
	SPL	SR	SPL	SR
OVRL [16]	12.3%*	32.8%*	7.0%	25.3%
CoW [21] (w/depth)	-	-	**6.3%**	11.1%
ZSON (ours)	12.6%	25.5%	4.8%	**15.3%**

(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% $\rightarrow$ 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% $\rightarrow$ 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% $\rightarrow$ 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]).

Method	Visual Encoder	Training Dataset	ImageNav (Gibson)		ObjectNav (Gibson)	
			SPL	SR	SPL	SR
ZER [20]	ResNet-9	Gibson	21.6%	29.2%	-	11.3%
Ours	ResNet-9	Gibson	**22.8%**	**33.3%**	**7.4%**	**15.3%**

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% $\rightarrow$ 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

[1] D. Batra, A. Gokaslan, A. Kembhavi, O. Maksymets, R. Mottaghi, M. Savva, A. Toshev, and E. Wijmans. Objectnav Revisited: On Evaluation of Embodied Agents Navigating to Objects. *arXiv preprint arXiv:2006.13171*, 2020.

[2] M. Savva, A. Kadian, O. Maksymets, Y. Zhao, E. Wijmans, B. Jain, J. Straub, J. Liu, V. Koltun, J. Malik, et al. Habitat: A platform for embodied ai research. In *ICCV*, 2019.

[3] A. Szot, A. Clegg, E. Undersander, E. Wijmans, Y. Zhao, J. Turner, N. Maestre, M. Mukadam, D. S. Chaplot, O. Maksymets, et al. Habitat 2.0: Training home assistants to rearrange their habitat. *NeurIPS*, 2021.

[4] F. Xia, A. R. Zamir, Z. He, A. Sax, J. Malik, and S. Savarese. Gibson Env: Real-World Perception for Embodied Agents. In *CVPR*, pages 9068–9079, 2018.

[5] E. Kolve, R. Mottaghi, W. Han, E. VanderBilt, L. Weihs, A. Herrasti, D. Gordon, Y. Zhu, A. Gupta, and A. Farhadi. AI2-THOR: An Interactive 3D Environment for Visual AI. *arXiv*, 2017.

[6] B. Talbot, D. Hall, H. Zhang, S. R. Bista, R. Smith, F. Dayoub, and N. Sünderhauf. BenchBot: Evaluating Robotics Research in Photorealistic 3D Simulation and on Real Robots, 2020.

[7] 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. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015.

[8] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niessner, M. Savva, S. Song, A. Zeng, and Y. Zhang. Matterport3D: Learning from RGB-D Data in Indoor Environments. In *ThreeDV*, 2017. MatterPort3D dataset license: `http://kaldir.vc.in.tum.de/matterport/MP_TOS.pdf`.

[9] I. Armeni, Z.-Y. He, J. Gwak, A. R. Zamir, M. Fischer, J. Malik, and S. Savarese. 3D Scene Graph: A Structure for Unified Semantics, 3D Space, and Camera. In *ICCV*, 2019.

[10] R. Ramrakhya, E. Undersander, D. Batra, and A. Das. Habitat-web: Learning embodied object-search strategies from human demonstrations at scale. In *CVPR*, 2022.

[11] D. S. Chaplot, D. Gandhi, A. Gupta, and R. Salakhutdinov. Object goal navigation using goal-oriented semantic exploration. In *NeurIPS*, 2020.

[12] J. Ye, D. Batra, A. Das, and E. Wijmans. Auxiliary tasks and exploration enable objectnav. In *ICCV*, 2021.

[13] O. Maksymets, V. Cartillier, A. Gokaslan, E. Wijmans, W. Galuba, S. Lee, and D. Batra. Thda: Treasure hunt data augmentation for semantic navigation. In *ICCV*, 2021.

[14] Y. Liang, B. Chen, and S. Song. SSCNav: Confidence-Aware Semantic Scene Completion for Visual Semantic Navigation. In *ICRA*, 2021.

[15] H. Luo, A. Yue, Z.-W. Hong, and P. Agrawal. Stubborn: A Strong Baseline for Indoor Object Navigation. *arXiv preprint arXiv:2203.07359*, 2022.

[16] K. Yadav, R. Ramrakhya, A. Majumdar, V.-P. Berges, S. Kuhar, D. Batra, A. Baevski, and O. Maksymets. Offline Visual Representation Learning for Embodied Navigation. *arXiv preprint arXiv:2204.13226*, 2022.

[17] K. Yadav, S. K. Ramakrishnan, A. Gokaslan, O. Maksymets, R. Jain, R. Ramrakhya, A. X. Chang, A. Clegg, M. Savva, E. Undersander, D. S. Chaplot, and D. Batra. Habitat challenge 2022. `https://aihabitat.org/challenge/2022/`, 2022.

[18] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning Transferable Visual Models from Natural Language Supervision. In *ICML*, 2021.

[19] S. K. Ramakrishnan, A. Gokaslan, E. Wijmans, O. Maksymets, A. Clegg, J. M. Turner, E. Undersander, W. Galuba, A. Westbury, A. X. Chang, M. Savva, Y. Zhao, and D. Batra. Habitat-matterport 3d dataset (HM3d): 1000 large-scale 3d environments for embodied AI. In *NeurIPS Datasets and Benchmarks Track*, 2021.

[20] Z. Al-Halah, S. K. Ramakrishnan, and K. Grauman. Zero Experience Required: Plug & Play Modular Transfer Learning for Semantic Visual Navigation. *arXiv preprint arXiv:2202.02440*, 2022.

[21] S. Y. Gadre, M. Wortsman, G. Ilharco, L. Schmidt, and S. Song. CLIP on Wheels: Zero-Shot Object Navigation as Object Localization and Exploration. *arXiv preprint arXiv:2203.10421*, 2022.

[22] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In *ICCV*, 2017.

[23] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In *CVPR*, 2016.

[24] A. Eftekhar, A. Sax, J. Malik, and A. Zamir. Omnidata: A Scalable Pipeline for Making Multi-Task Mid-Level Vision Datasets From 3D Scans. In *ICCV*, 2021.

[25] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging Properties in Self-Supervised Vision Transformers. In *ICCV*, 2021.

[26] E. Wijmans, A. Kadian, A. Morcos, S. Lee, I. Essa, D. Parikh, M. Savva, and D. Batra. DD-PPO: Learning Near-Perfect PointGoal Navigators from 2.5 Billion Frames. In *ICLR*, 2019.