

Instance-aware Exploration-Verification-Exploitation for Instance ImageGoal Navigation

Xiaohan Lei Min Wang Wengang Zhou Li Li Houqiang Li

Department of Electrical Information Engineering, University of Science and Technology of China

Abstract

As a new embodied vision task, Instance ImageGoal Navigation (IIN) aims to navigate to a specified object depicted by a goal image in an unexplored environment. The main challenge of this task lies in identifying the target object from different viewpoints while rejecting similar distractors. Existing ImageGoal Navigation methods usually adopt the simple Exploration-Exploitation framework and ignore the identification of specific instance during navigation. In this work, we propose to imitate the human behaviour of “getting closer to confirm” when distinguishing objects from a distance. Specifically, we design a new modular navigation framework named Instance-aware Exploration-Verification-Exploitation (IEVE) for instance-level image goal navigation. Our method allows for active switching among the exploration, verification, and exploitation actions, thereby facilitating the agent in making reasonable decisions under different situations. On the challenging HabitatMatterport 3D semantic (HM3D-SEM) dataset, our method surpasses previous state-of-the-art work, with a classical segmentation model (0.684 vs. 0.561 success) or a robust model (0.702 vs. 0.561 success). Our code will be made publicly available at <https://anonymous.4open.science/r/IEVE-0CC3>.

1. Introduction

Embodied navigation is an emerging computer vision task, where an agent utilizes visual sensing to actively interact with its surroundings and execute navigation tasks [2, 4]. Over the past decade, substantial advancements have been witnessed in the field of embodied visual navigation, driven by the emergence of large-scale photorealistic 3D scene datasets [6, 41, 48, 53, 57], the implementation of advanced simulators [24, 46, 49, 53], the development of deep learning algorithms [5, 19, 38, 47], and so on.

The ability for an agent to navigate its environment and accomplish tasks without human intervention is essential for the advancement of embodied visual navigation. One common setting for tasking an agent is to give it natural



Figure 1. **Instance ImageGoal Navigation** tasks an agent with navigating to a particular object instance described by the goal image. The agent is initially in an unexplored environment at t_1 . While exploring the environment, it encounters a “bed” similar to the target at t_2 and discerns their differences. Eventually, at t_3 , agent finds the “bed” described in the goal image.

language instructions, such as “check if my laptop is on the chair.” However, this setting becomes confusing when there are multiple chairs in the house. To overcome this challenge, Krantz *et al.* [26] propose the Instance ImageGoal Navigation task (IIN), which involves providing the agent with an image captured from goal camera that depicts a specific instance. The agent is then required to navigate to the location of the instance depicted in the image, as shown in Fig. 1. By utilizing the visual information, the agent is expected to avoid confusion caused by multiple similar distractors of the same class.

The most relevant task with IIN is ImageGoal Navigation [1, 9, 17, 33, 35, 56, 63], which has been a popular topic in embodied visual navigation research. The difference between IIN and ImageGoal Navigation lies in their requirements for navigation success. In other words, IIN demands that the agent has the ability to recognize the same object from different viewpoints, while ImageGoal Navigation requires maximizing the similarity between the agent’s observations and the goal image. Previous ImageGoal navigation requires targeting random goal images captured by cameras that share the same parameters like height, look-at-angle, and field-of-view as the agent camera. Some approaches [28, 52] employ local feature matching algorithms and make binary decisions based on the current observation. They typically expect the agent to accurately determine whether the current observation matches the goal image. However, it is extremely challenging to enable the agent to correctly identify the target object from a considerable distance. In addition, they treat the decision-making process as a one-shot task, perceiving it as a match between the current observation and the target, while overlooking the judgment process across the temporal dimension.

To address the above limitations, we propose a novel IIN framework, *i.e.*, Instance-aware Exploration-Verification-Exploitation (IEVE), for Instance ImageGoal Navigation. Inspired by the human behavior of “getting closer to confirm” when recognizing distant objects, we formulate the task of determining whether an object matches the one in the goal image as a sequential decision problem. In addition, we design a novel matching function that relies not only on the current observation and goal image but also on the Euclidean distance between the agent and the potential target. We categorize the targets into confirmed target, potential target, and no-target (exploration), and allow the agent to actively choose among these three targets. After an efficient training process of our framework, our result on the IIN task outperforms previous state-of-the-art methods, achieving 0.684 *Success* with a classical instance segmentation model [18], and 0.702 with a robust model [51].

2. Related Work

In recent years, there has been great research interest in the field of embodied vision and active agents. Embodied agents are required to effectively and creatively resolve complex tasks and interact with the physical environment. Notably, the navigation task plays a pivotal role in embodied vision. Numerous researchers have devoted their efforts to exploring different forms of navigation problems, encompassing various task goals and input formats. Specifically, there are five main categories of visual navigation problems: Point Goal Navigation [2, 8, 45], Object Goal Navigation [4, 14, 15, 45, 61], Visual Language Navigation [3, 16, 20–22, 25, 27, 30, 30, 59], ImageGoal Navigation

[1, 9, 17, 29, 33, 35, 56, 63], and Instance ImageGoal Navigation [28]. According to the learning framework design, existing visual navigation methods are typically divided into two categories: modular and end-to-end approaches.

End-to-end navigation policies are commonly optimized with improved visual representations [37, 58], auxiliary tasks [60], carefully reward-shaping [34] and data augmentation techniques [34]. Visual representations can be improved with fine-tuned segmentation model [37] and object relation graphs [12, 13, 36, 39, 62]. Prior works also learn fine-tuning visual encoder with Self-Supervised Learning (SSL) [40, 55], while some researchers employ imitation learning approaches, such as Ramrakhyia *et al.* [43], gathering data from large-scale human annotations. Although achieving promising navigation success in simulator, end-to-end methods often suffer high computation complexity in training and generalizability, which may degrade the success of navigation in diverse environments.

To address these challenges, modular methods have been proposed to solve navigation tasks with separate modules, including perception, mapping, path planning, *etc.* Chaplot *et al.* [7] address the Object Goal Navigation problem using semantic mapping and exploration. Majumdar *et al.* [33] adopt the zero-shot learning and inference approach with the use of CLIP. Rmakrishnan *et al.* [42] propose a potential field function to tackle the exploration problem in Object Goal Navigation. Mezghan *et al.* [35] and Savinov *et al.* [44] construct augmented memory mechanisms in ImageGoal Navigation. Some approaches [28, 52] employ local feature matching algorithms and make binary decisions based on the current observation for ImageGoal Navigation. In general, modular-based methods typically exhibit excellent scalability with lower training cost.

In this work, we focus on IIN task, which is a newly formulated but more real navigation task. It is relevant with the classical ImageGoal Navigation task. ImageGoal Navigation is commonly studied in a previous-unseen environment where goal images are randomly sampled sharing the same parameter with the agent’s camera [63]. Many approaches study the ImageGoal Navigation task with RL to learn policies that directly map observations to action [1, 10, 55, 56]. Yadav *et al.* [55] postulate that SSL for visual representation greatly contributes to the success of the end-to-end navigation policy training. Choi *et al.* [10] attribute the success of RL training to the carefully reward shaping. Chaplot *et al.* [9] propose a topological mapping system to address the ImageGoal navigation problem, as well as Kim *et al.* [23]. Kwon *et al.* [29] propose a novel type of map for ImageGoal Navigation, a renderable neural radiance map.

Unlike ImageGoal Navigation, IIN [26] focuses on distinguishing different similar instances and being able to identify the object in the goal image at different viewpoints. On the other hand, ImageGoal Navigation pays more at-

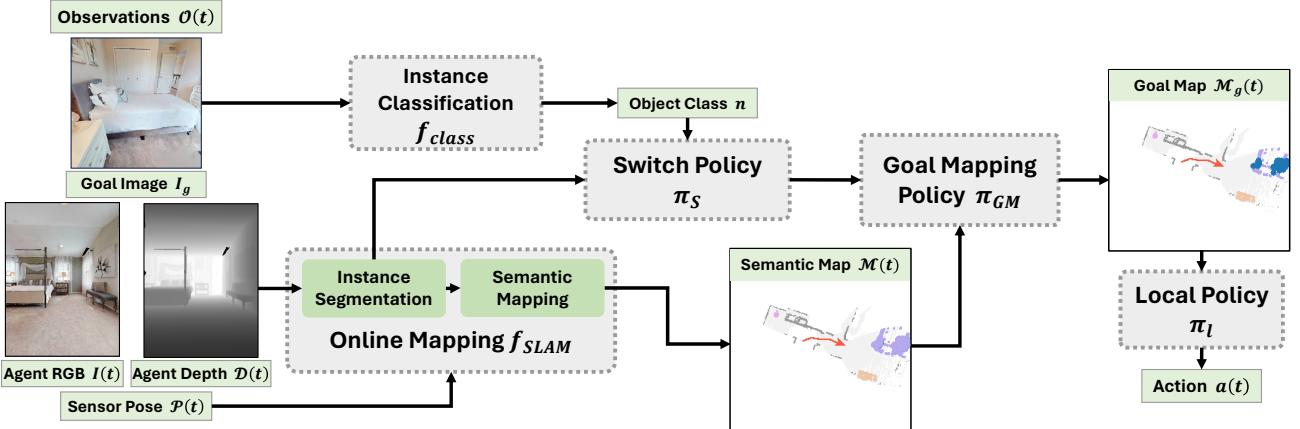


Figure 2. Framework Overview. Our model consists of five main components. Instance Classification f_{class} predicts the object’s class in goal image n . Online Mapping f_{SLAM} uses RGB-D and sensor pose reading $\mathcal{P}(t)$ to construct a semantic map $\mathcal{M}(t)$ of the environment. The Switch Policy π_S and Goal Mapping Policy π_{GM} are interconnected. The Switch Policy π_S determines the output of the Goal Mapping Policy’s goal map $\mathcal{M}_g(t)$ based on its input judgments (whether a potential target exists and whether the potential target is confirmed). Once the goal map $\mathcal{M}_g(t)$ is determined, the Local Policy π_l is used to determine the action $a(t)$ taken by the agent at the current timestep.

tention to maximizing the similarity between observations and the goal image within an episode. To be more specific, the goal camera in IIN task is disentangled from the agent’s camera on the sampled parameters such as height, look-at-angle, and field-of-view reflect the realistic use case of a user-supplied goal image. Additionally, previous state-of-the-art work employs the confidence score for object re-identification. Differently, we propose a new switchable strategy to meticulously examine a currently visible object. Our key insight lies in treating the subtask of target confirmation as a sequential decision problem and designing a new matching function, which enables the agent to more accurately distinguish target objects from misleading ones.

3. Method

We devise a new framework, Instance-aware Exploration-Verification-Exploitation (IEVE) for Instance ImageGoal Navigation, which consists of five key modules, including Instance Classification, Online Mapping, Switch Policy, Goal Mapping Policy and Local Policy, as depicted in Fig. 2. In the following subsections, we will first introduce the preliminaries in Sec. 3.1, and sequentially elaborate the five modules of our framework.

3.1. Preliminaries

We follow the embodiment setting of Hello Robot Stretch¹. The agent in the simulation environment is a rigid-body, zero-turn-radius cylinder of height 1.41m and radius 0.17m. The forward-facing RGB-D camera is mounted at a height of 1.31m. At each timestep t , the agent’s observation \mathcal{O}_t consists of an egocentric RGB image \mathcal{I}_t , depth image $\mathcal{D}(t)$,

goal image \mathcal{I}_g and sensor pose reading $\mathcal{P}(t)$. Camera parameters of \mathcal{I}_t and \mathcal{I}_g like height, look-at-angle and field-of-view are not the same. Specifically, \mathcal{I}_t is a 640×480 image while \mathcal{I}_g is a 512×512 image.

3.2. Instance Classification

At the onset of each episode, the Instance Classification model will first process \mathcal{I}_g and output an object class label n . We adopt two different types of models to achieve instance classification, including 1) classification model: we use the goal images on the train split of HM3D-SEM [57] to finetune the image classification model, *i.e.*, Swin Transformer [32], pretrained on ImageNet [11]. It may face the inherent semantic ambiguity of Ground Truth labels in the HM3D-SEM, such as the difficulties in distinguishing between the class “sofa” and “chair”. 2) instance segmentation model: we use the semantic label of the most central segmented instance in the goal image as the classification result. This approach can maintain semantic consistency without being affected by Ground Truth labels.

3.3. Online Mapping

At each time step t , the agent will utilize the egocentric view to construct the current semantic map $\hat{\mathcal{M}}(t)$ within its local coordinate system. The RGB frame $\mathcal{I}(t)$ is first processed through instance segmentation (Mask RCNN [18]), yielding the output $\mathcal{I}_{seg}(t)$ with dimensions of $(B, N + 3, H, W)$, denoting batch size, the number of classes N plus RGB channels of $\mathcal{I}(t)$, frame height, and frame width. Simultaneously, the depth image $\mathcal{D}(t)$ is transformed into a pointcloud $P_c(t)$ using the camera’s intrinsic matrix \mathbf{T}_{int} , followed by the voxelization of the pointcloud. Subsequently, the feature points of $\mathcal{I}_{seg}(t)$ are scattered into the

¹<https://hello-robot.com/stretch-2>

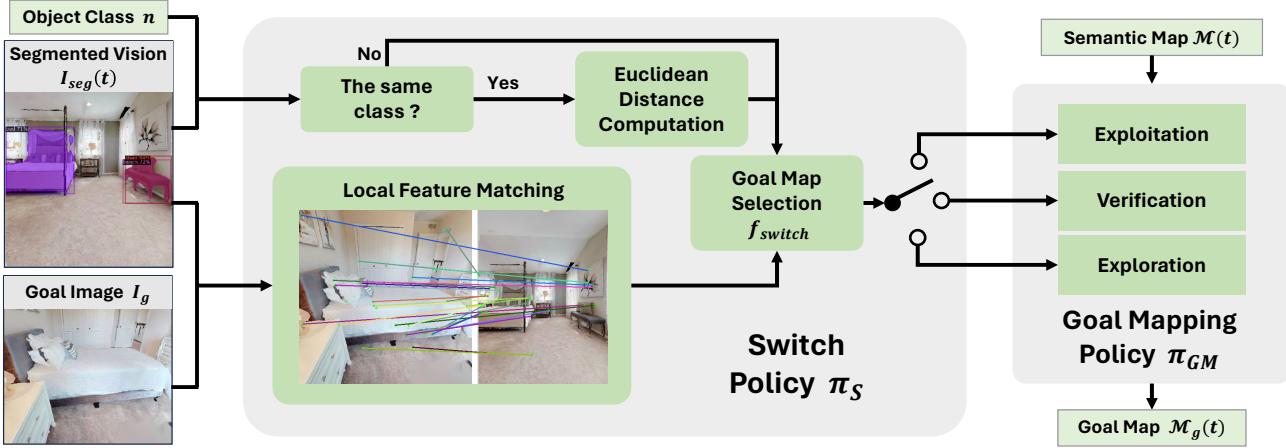


Figure 3. Switch Policy and Goal Mapping Policy. The Goal Map Selection module f_{switch} receives two inputs: the shortest Euclidean distance between the agent and the potential target (or non-existence), and the number of matched keypoints. These inputs are mapped to a selection signal using Goal Map Selection function f_{switch} . At each timestep, the Switch Policy π_S will choose one and only one of the three parallel modules of Goal Mapping Policy π_{GM} .

voxels, resulting in the overall local map construction process being formulated as:

$$\hat{m}(t) = \text{Scatter}(\text{Voxel}(\mathbf{T}_{int}\mathcal{D}(t)), \mathcal{I}_{seg}(t)). \quad (1)$$

With the constructed semantic map $\hat{m}(t)$ within local coordinate system, agent will subsequently conduct max pooling with the previous global semantic map to obtain the global semantic map $M(t)$ at timestep t as follows:

$$M(t) = \text{MaxPooling}(M(t-1), \mathbf{T}_{lg}\hat{m}(t)), \quad (2)$$

where \mathbf{T}_{lg} denotes the transformation matrix from the agent's local frame to the world's global coordinate system. The size of $M(t)$ is specified as $(B, 2+N, H_m, W_m)$, where the first channel represents the explored area of the entire environment, the second channel defines the non-traversable area, and the last N channels signify the location of each object class.

3.4. Switch Policy

At each time step t , Switch Policy actively selects the functional goal map from the three parallel sub-modules in Goal Mapping Policy. We summarize the switch process as a function with segmented depth image $\mathcal{I}_{seg}(t, n) \cdot \mathcal{D}(t)$, the number of matched keypoints ω using local feature matching algorithm and the minimum Euclidean distance d between the current position of the agent and the potential target (if exists). This can be described as follows:

$$S = f_{switch}(\mathcal{I}_{seg}(t, n) \cdot \mathcal{D}(t), \omega, d), \quad (3)$$

where S indicates which goal mapping policy to be selected, depicted in Fig. 3. To obtain the specific expression of f_{switch} , we construct an Instance Re-Identification

dataset as depicted in Fig. 6 to illuminate the influence of distance d and the number of matched keypoints w on the goal object's re-identification.

The Goal Map Selection function f_{switch} aligns with human experience in object verification. f_{switch} sets objects of the same category as potential targets. At a larger distance to make accurate decisions, a higher threshold on number of matched keypoints is set to convert a potential target into a determined target as depicted in Fig. 4. As the distance decreases, the upper threshold also degrades continuously. At closer distance, the agent's judgment of the object becomes more reliable, so when the number of matched keypoints falls below the lower threshold, it can be very reliably determined that the potential target is not the goal target, thereby switching to the Exploration Policy.

Local Feature Matching. We employ a keypoint-based re-identification method to help in the selection of goal mapping policy in Eq. (3). Conditioned on the goal image \mathcal{I}_g and egocentric image $\mathcal{I}(t)$, we extract the pixel-wise (x, y) coordinates of the keypoints $K_t \in \mathbb{R}^{n \times 2}$ and their associated feature descriptors $V_t \in \mathbb{R}^{n \times 256}$. These features are extracted utilizing DISK [50], an end-to-end local feature learning algorithm that utilizes policy gradient. Likewise, the same process is repeated with the goal image, resulting in $K_g \in \mathbb{R}^{m \times 2}$ and $V_g \in \mathbb{R}^{m \times 256}$. The feature extraction processes are formulated as follows:

$$(K_t, V_t) = \text{DISK}(\mathcal{I}(t)), \quad (K_g, V_g) = \text{DISK}(\mathcal{I}_g). \quad (4)$$

Subsequently, the matched pairs $(\hat{K}_t, \hat{K}_g) \in \mathbb{R}^{\omega \times 2}$ are computed using LightGlue [31], a deep neural network that matches sparse local features across image pairs. The feature matching process is formulated as follows:

$$(\hat{K}_t, \hat{K}_g) = \text{LightGlue}((K_t, V_t), (K_g, V_g)). \quad (5)$$

Subsequently, the number of matched points ω is utilized as an adaptive threshold for the verification and exploitation goal selection.

3.5. Goal Mapping Policy

Depicted in Fig. 3, Goal Mapping Policy module is divided into three parallel sub-modules, which will function at different timesteps. In other words, at each timestep, only one sub-module will function. More specifically, given the predicted goal object’s class n and segmented vision $\mathcal{I}_{seg}(t)$, if the mask $\mathcal{I}_{seg}(t, n)$ has any value greater than zero, then the Exploitation or Verification Policy is activated. They project the masked depth image $\mathcal{I}_{seg}(t, n) \cdot \mathcal{D}(t)$ to current semantic map. Similar to previous Online Mapping module, the two goal mapping policies in local coordinate system can be summarized as follows:

$$\hat{m}_g(t) = \text{Voxel}(\mathbf{T}_{int}\mathcal{I}_{seg}(t, n) \cdot \mathcal{D}(t)), \quad (6)$$

where $\hat{m}_g(t)$ indicates the goal map in the local coordinate system. Then we transform $\hat{m}_g(t)$ into the global coordinate system, $\mathcal{M}_g(t) = \mathbf{T}_{lg}\hat{m}_g(t)$. Although with the same goal map formulas, Exploitation and Verification policy represent the targets with different confidence. Specifically, Exploitation policy refers to the confirmed target, which will be kept and used for the rest navigation process. In contrast, Verification policy represents potential target, which will be further checked by local feature matching in the Switch policy module.

The Exploration Policy use a convolutional neural network to predict a waypoint $g_{rl}(t) = \pi_{rl}(\mathcal{M}(t)|\theta_{rl})$, where θ_{rl} are parameters of Exploration Policy. We train it with RL to maximize coverage of the environment while also successfully employing the *stop* action upon encountering a correct oracle facing the goal object. The goal map $\mathcal{M}_g(t)$ is constructed with the value at position $g_{rl}(t)$ set as 1.

3.6. Local Policy

When Switch Policy selects one of three goal mapping policies, agent adopts Local Policy to infer next action in accordance with the semantic map $\mathcal{M}(t)$, and the goal map, $\mathcal{M}_g(t)$. Specifically, Fast Matching Method (FMM) is employed to generate a distance field, which establishes the shortest distance between a traversible point and the selected goal map. A subset of the distance field is chosen within the agent’s feasible range, and a waypoint is selected, ensuring that it does not intersect with existing obstacles and maintains the local minimum within the subset. With the selected waypoint, the agent can readily calculate an action based on the angle and distance from its current state. We also add a visited trajectory mechanism for the agent to avoid been stuck by invisible obstacles, which will be discussed in the supplementary materials. In our framework, only Instance Classification and Exploration Policy require

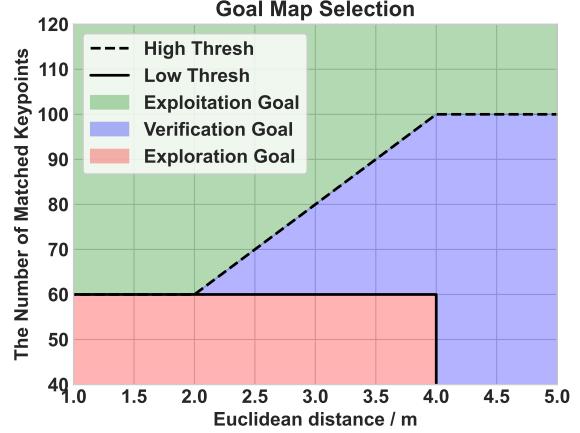


Figure 4. Goal Map Selection function f_{switch} with respect to the Euclidean distance from agent to the potential target and the number of matched keypoints. Best viewed in color.

Model	Validation	
	Success \uparrow	SPL \uparrow
RL Baseline [26]	0.083	0.035
OVRL-v2 ImageNav [55]	0.006	0.002
OVRL-v2 IIN [55]*	0.248	0.118
Mod-IIN [28]	0.561	0.233
IEVE Mask RCNN (ours)	0.684	0.241
IEVE InternImage (ours)	0.702	0.252

Table 1. Comparison between our IEVE and prior methods on IIN task. “*” means we finetune OVRL-v2 [55] on the IIN task. IEVE outperforms the baseline model (row 1) with up to 8.75x increase in *Success* and outperforms a state-of-the-art model [28] (Mod-IIN, row 4) from 0.561 to 0.684 with a lightweight segmentation model (Mask RCNN) on *Success*, and higher *Success* of 0.702 with a strong backbone (InternImage [51]).

few training, underscoring the computation efficiency of our design.

4. Experiment

4.1. Experiment Setup

Dataset. For our experiment, we employ the Habitat [49] simulator to conduct our experiments. The definition of the Instance ImageGoal Navigation [26] task utilized in the Habitat Navigation Challenge 2023² serves as our primary reference point. The episode dataset proposed by Krantz et al. [28] for Instance ImageGoal Navigation serves as our key data source. The scene dataset utilized in this study stems from the Habitat-Matterport3D with semantic annotations (HM3D-SEM) [54], which encompasses a total of 216 scenes. These scenes are split into three distinct subsets for training, validation, and testing purposes, consist-

²<https://aihabitat.org/challenge/2023/>

ing of 145/36/35 scenes, respectively. Additionally, the episode dataset has been partitioned into four subsets for training, validation, testing, and testing under a modified challenge regime, comprising 7,056K/1K/1K/1K episodes respectively. We train our agent on the train split and evaluate it on validation split. The object depicted by the goal image belongs to the following six categories: { *chair*, *couch*, *plant*, *bed*, *toilet*, *television* }. On the validation subset, a total of 795 unique object instances have been observed.

Action Space. Adhering to the task setting of the Habitat Navigation Challenge 2023, we employ the continuous action space. The action space encompasses four actions: *linear velocity*, *angular velocity*, *camera pitch velocity*, *velocity stop*, each of which accepts values ranging between -1 and 1 and is appropriately scaled according to the respective configurations. Our linear speed is capped at a maximum of $35\text{cm}/\text{frame}$ and angular velocity at $60^\circ/\text{frame}$.

Metrics. We evaluate our model with both success and efficiency. We report Success Rate (*Success*), Success rate weighted by normalized inverse Path Length (*SPL*). *Success* is True if the agent calls *velocity stop* action within 1.0m Euclidean distance of the goal object and the object is oracle-visible by turning or looking up and down. *SPL* is an efficiency measure defined in [2].

4.2. Comparison with State-of-the-art Methods

We undertake a comprehensive evaluation of our proposed model against various baselines and the SOTA work.

End-to-End RL Agent. The baseline model for the IIN task is trained in a fully-fledged end-to-end fashion. This involves inputting both an egocentric rgb and depth image as well as a pose into the model, which is then processed by a visual encoder and a recurrent neural network. The network output directly determines the agent’s action. The agent is trained from scratch utilizing Proximal Policy Optimization (PPO) [47]. Our method exhibits 8.24x higher *Success* in Val (row 5 vs. row 1 in Tab. 1). Although it takes a staggering 3.5 billion steps for RL agent model to achieve convergence, the generalization capability of the agent remains unsatisfactory.

OVRL-v2. Offline Visual Representation Learning v2 (OVRL-v2 [55]) was proposed by Karmesh *et al.*, who introduced a novel model-free, end-to-end navigation policy that emphasizes the significance of self-supervised pretraining of visual encoders and introduces the concept of Compression Layer (CL). Employing a ViT + LSTM backbone, this policy exhibits state-of-the-art performance in ImageGoal Navigation and approaches the state-of-the-art in ObjectGoal Navigation. Due to the task disparity between ImageGoal Navigation and Instance ImageGoal Navigation, we fine-tune the pretrained policy for our specific task.

The original OVRL-v2 is trained under the setting of the ImageGoal Navigation task. Evaluating OVRL-v2’s per-

Classification	Perception	Model Variation		Validation	
		Switch	Exploration	<i>Success</i> \uparrow	<i>SPL</i> \uparrow
Mask RCNN	Mask RCNN	EVE	RL	0.666	0.235
Mask RCNN	Mask RCNN	EVE	Frontier	0.676	0.182
Mask RCNN	Mask RCNN	EVE	Random	0.654	0.233
Mask RCNN	Mask RCNN	EE	RL	0.590	0.221
InternImage	InternImage	EVE	RL	0.702	0.252
SwinTrans	Mask RCNN	EVE	RL	0.684	0.241
SwinTrans	Mask RCNN	EVE	Frontier	0.628	0.180
SwinTrans	Mask RCNN	EVE	Random	0.636	0.236
SwinTrans	Mask RCNN	EE	RL	0.620	0.216
SwinTrans	InternImage	EVE	RL	0.676	0.238
GroundTruth	Mask RCNN	EVE	RL	0.682	0.238
GroundTruth	Oracle	-	RL	0.850	0.399

Table 2. Variants of our method with different settings of instance classification, perception, switch policy and exploration policy. Perception means the Instance Segmentation model variations in Online Mapping module. “EE” refers to the exploration-exploitation policy, which establishes a fixed threshold for decision-making. “EVE” represents our Exploration-Verification-Exploitation Switch policy. The last row gives the upper bound of our method, where with ground truth instance segmented observations, the agent can identify the specified object instance as soon as the object becomes visible.

formance without finetuning for Instance ImageGoal Navigation reveals suboptimal results (0.006 *Success* row 2 in Tab. 1). Several factors impact the performance: a shift in scene datasets (Gibson [53] vs. HM3D-SEM [57]), changes in embodiment (Locobot vs. Stretch), and disparity in goal destinations (image source vs. image subject). When specifically finetuned for Instance ImageGoal Navigation (row 3 in Tab. 1) on HM3D-SEM, OVRL-v2 yields improved performance rates at 0.248 *Success*. Our approach with Mask RCNN exhibits 0.436 higher *Success* in Val (row 5 vs. row 3 in Tab. 1), and 0.454 higher *Success* with InternImage (row 6 vs. row 3 in Tab. 1).

Mod-IIN. Mod-IIN [28] decomposes Instance ImageGoal Navigation task into exploration, goal re-identification, goal localization and local navigation. It outperforms previous state-of-the-art end-to-end learned policies without fine-tuning. Mod-IIN achieves 0.561 *Success* and 0.233 *SPL* (row 4 in Tab. 1), and we surpass their results in *Success* with a substantial margin of 0.123 by attaining 0.684 *Success* and a marginal improvement in *SPL* with the help of a classical visual backbone trained with Mask RCNN [18] (row 5 in Tab. 1). Additionally, with the current state-of-the-art Semantic Segmentation model InternImage [51], our results (row 6 in Tab. 1) are notably superior with an impressive score of 0.702 *Success* and 0.252 *SPL*.

4.3. Ablation Study

Instance Classification. Our instance classification method falls into two categories. One is to fine-tune the pretrained SwinTransformer [32] on the goal images of the HM3D-SEM training set. The other method is to obtain the seman-

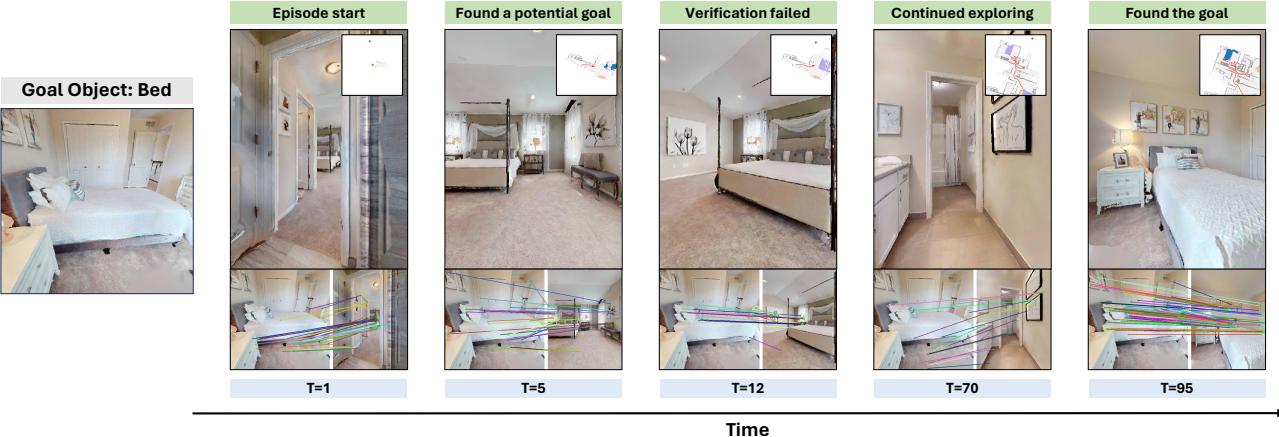


Figure 5. Qualitative example of our TEVE agent performing the Instance ImageGoal Navigation task in the Habitat simulator. Agent is initialized at $T = 1$, and finds a potential target at $T = 5$. After carefully evaluating the potential target from $T = 5$ to $T = 12$, the agent proceeds with its exploration. Finally, agents identifies the goal bed instance at $T = 95$.



Figure 6. **Instance Re-Identification Dataset.** Based on the object depicted in the goal image, we sample positive and negative samples with varying Euclidean distance which represents the difficulty level of instance re-identification. For each difficulty level, the left side shows the ground truth segmentation results and the right side shows the RGB images.

tic category of the most central segmentation result of the goal image using the segmentation model. Ablation experiments conducted on these two methods demonstrate that SwinTrans is helpful for Mask RCNN (row 6 vs. row 1 in Tab. 2), but it actually suppresses the performance of InternImage (row 10 vs. row 5 in Tab. 2). We believe this is because the InternImage segmentation model has a larger parameter size, better zero-shot capabilities on this dataset, and stronger semantic labeling capabilities.

Exploration Policy. To scrutinize the extent to which exploration policy exerts a meaningful influence on final outcomes, we separately train our framework with three different exploration policies: a random exploration policy, an RL exploration policy, and a frontier exploration policy. Each of these three policies generates an exploration goal map every ten local steps. The first random policy involves taking the semantic map as input and randomly generating a goal map. The second policy, trained through Reinforcement Learning, seeks to maximize coverage of the environ-

ment while also encouraging the *velocity stop* action upon encountering a correct oracle facing the goal object. The frontier exploration policy, in contrast, endeavors to delineate the boundaries of the currently explored and free regions of the map, thereby constructing a goal map that encourages exploration. The implementation details of exploration policy are discussed in the supplementary materials.

From Tab. 2, the performance of RL (row 1 and row 6 in Tab. 2) and random (row 3 and row 8 in Tab. 2) exploration policy on the *SPL* significantly exceeds that of Frontier exploration policy (row 2 and row 7 in Tab. 2). This disparity in performance is due to the fact that Frontier Exploration requires exploring every corner of a given environment, which is time-consuming and unnecessary in most cases. Specifically, the agent should focus on gaining a comprehensive understanding of the most salient aspects of a small room, which facilitates the identification of the target object.

Instance Segmentation. Initially, we employ the Mask RCNN instance segmentation model on the Online Mapping module. In order to explore the influence of the visual model on the framework, we have employed the current state-of-the-art semantic segmentation approach: InternImage [51] finetuned on the COCO dataset. Furthermore, in order to ascertain the upper bound of our method, we have employed the ground truth labeled instance class and the ground truth instance segmentation to conduct comprehensive evaluations of the *Success* and *SPL* rates. This entails the use of a local policy for the purpose of goal map-to-action, where the agent is directly capable of ascertaining whether the goal object is present in the current view while exploring the environment.

From Tab. 2, we can observe that state-of-the-art semantic segmentation model InternImage (row 5 in Tab. 2) indeed benefits navigation, and the ground truth segmented

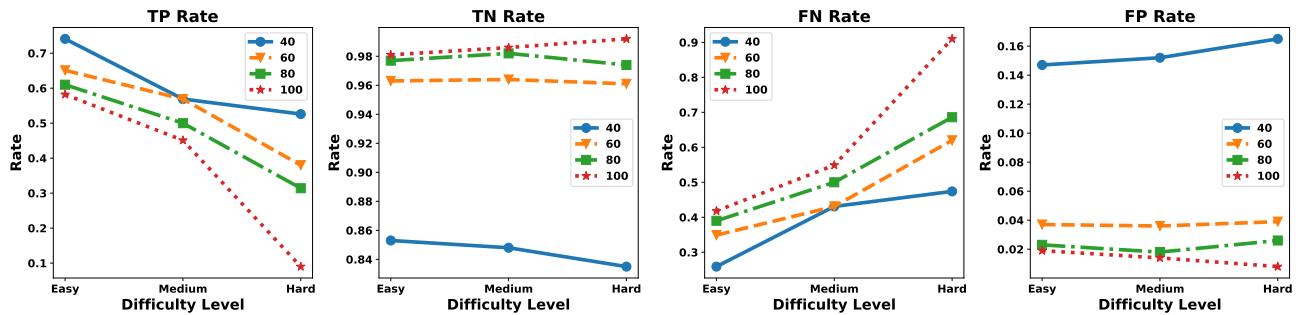


Figure 7. The TP, TN, FN and FP rates vary with changes in the threshold and difficulty levels (representing the closest Euclidean distance between the agent and the potential target). Threshold represents the threshold for the number of matched keypoints that determines whether two images are considered to be the same instance. TP decays rapidly as the difficulty increases, indicating that it becomes increasingly challenging for the agent to make correct judgments at a larger distance.

observation (last row in Tab. 2) gives the upper bound of navigation. Specifically, more precisely segmented observation from InternImage gives higher metrics both in *Success* and *SPL*. This is due to the more precise goal localization and less wrong segmentation results.

Switch Policy. We conduct a comparative analysis of Switch policy. We innovatively incorporate a verification policy into the decision-making process of the agent, and through ablation experiments (row 1 vs. row 4 and row 6 vs. row 9 in Tab. 2), we demonstrate its effectiveness and significance. The implementation details of the Exploration-Exploitation Policy will be presented in supplementary materials. As depicted in Tab. 2, our EVE policy (row 1, row 6 in Tab. 2) outperforms Exploration-Exploitation (EE) Policy (row 4, row 9 in Tab. 2) by a substantial margin both on *Success* and *SPL*. This suggests that the EVE policy provides the agent with greater opportunities to discern the current observation from the goal image, rather than arbitrarily assigning a false label to the current observed object in the EE policy. Depicted in Fig. 5, agent finds the potential target at $T = 5$ and employ verification policy. After carefully examining the similar object, the agent determines that the object is not the target and actively switches back to Exploration Policy.

Instance Re-Identification. To assess our Switch module and identify optimal hyperparameters of the Goal Map Selection module f_{switch} , we have constructed a comprehensive dataset, as depicted in Fig. 6. This dataset is created based on the HM3D-SEM train/val split. Anchor images are defined as goal images that depict specific object instances, which do not share the same parameters as agent’s camera.

For each anchor goal image, we sample 10 positive and negative images using the agent’s camera. Positive images are defined as observations where the target instance is visible to the oracle, while negative images are randomly sampled from the same scene and do not contain the specified goal instance. We have divided the dataset into three distinct difficulty levels: *easy*, *medium*, *hard*, as shown in Fig. 6. Different difficulty levels represent different Euclidean dis-

tances from the target to current viewpoint. The further implementation details will be discussed in the supplementary materials. To demonstrate the capabilities of our Switch module, we employ the train split to determine hyperparameters and subsequently evaluate performance on the val split. Notably, the train and val splits are two distinct sets, which ensure that the agent has never encountered any scene or object instance present in the train split.

As depicted in Fig. 7, it is evident that with a fixed threshold of 60, the TP values for the difficulty level decrease significantly. It becomes more evident in the hard level with a threshold of 100 that yields TP rate close to 0. This highlights the importance of careful examination and decision-making based on the utilization of local matching feature algorithms to differentiate between similar objects. By meticulously examining the possible object instance, the agent is able to confidently arrive at a decision. We believe this explains why our model significantly surpasses previous state-of-the-art research in terms of *Success*.

5. Conclusion

We propose an innovative framework of Instance-aware Exploration-Verification-Exploitation for Instance ImageGoal Navigation. We rethink previous work that decouple the decision-making process into an Exploration-Exploitation paradigm. By considering human behavior in the “approach-verify” phase of identifying potential targets, we propose a new decision-making paradigm: Exploration-Verification-Exploitation. Our experimental results demonstrate that the Exploration-Verification-Exploitation paradigm effectively aids agents in their planning and decision-making process. Our proposed model significantly outperforms existing methods on the Instance ImageGoal Navigation task (0.684 vs. 0.561 *Success*) with a lightweight instance segmentation model. In the future, we will focus on enhancing the context of visual reasoning for more efficient exploration, and extend the application of the three-stage planning and decision-making paradigm to a broader range of tasks of embodied vision.

References

- [1] Ziad Al-Halah, Santhosh Kumar Ramakrishnan, and Kristen Grauman. Zero experience required: Plug & play modular transfer learning for semantic visual navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17031–17041, 2022. [2](#)
- [2] Peter Anderson, Angel Chang, Devendra Singh Chaplot, Alexey Dosovitskiy, Saurabh Gupta, Vladlen Koltun, Jana Kosecka, Jitendra Malik, Roozbeh Mottaghi, Manolis Savva, et al. On evaluation of embodied navigation agents. *arXiv preprint arXiv:1807.06757*, 2018. [1, 2, 6](#)
- [3] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3674–3683, 2018. [2](#)
- [4] Dhruv Batra, Aaron Gokaslan, Aniruddha Kembhavi, Oleksandr Maksymets, Roozbeh Mottaghi, Manolis Savva, Alexander Toshev, and Erik Wijmans. Objectnav revisited: On evaluation of embodied agents navigating to objects. *arXiv preprint arXiv:2006.13171*, 2020. [1, 2](#)
- [5] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9650–9660, 2021. [1](#)
- [6] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017. [1](#)
- [7] Devendra Singh Chaplot, Dhiraj Gandhi, Abhinav Gupta, and Ruslan Salakhutdinov. Object goal navigation using goal-oriented semantic exploration. In *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, 2020. [2](#)
- [8] Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, and Ruslan Salakhutdinov. Learning to explore using active neural slam. *arXiv preprint arXiv:2004.05155*, 2020. [2](#)
- [9] Devendra Singh Chaplot, Ruslan Salakhutdinov, Abhinav Gupta, and Saurabh Gupta. Neural topological slam for visual navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. [2](#)
- [10] Yunho Choi and Songhwai Oh. Image-goal navigation via keypoint-based reinforcement learning. In *Proceedings of the International Conference on Ubiquitous Robots (UR)*, pages 18–21, 2021. [2](#)
- [11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009. [3](#)
- [12] Raphael Druon, Yusuke Yoshiyasu, Asako Kanezaki, and Alassane Watt. Visual object search by learning spatial context. *IEEE Robotics and Automation Letters*, 5(2):1279–1286, 2020. [2](#)
- [13] Heming Du, Xin Yu, and Liang Zheng. Learning object relation graph and tentative policy for visual navigation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 19–34. Springer, 2020. [2](#)
- [14] Heming Du, Lincheng Li, Zi Huang, and Xin Yu. Object-goal visual navigation via effective exploration of relations among historical navigation states. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2563–2573, 2023. [2](#)
- [15] Samir Yitzhak Gadre, Mitchell Wortsman, Gabriel Ilharco, Ludwig Schmidt, and Shuran Song. Cows on pasture: Baselines and benchmarks for language-driven zero-shot object navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23171–23181, 2023. [2](#)
- [16] Chen Gao, Xingyu Peng, Mi Yan, He Wang, Lirong Yang, Haibing Ren, Hongsheng Li, and Si Liu. Adaptive zone-aware hierarchical planner for vision-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14911–14920, 2023. [2](#)
- [17] Meera Hahn, Devendra Singh Chaplot, Shubham Tulsiani, Mustafa Mukadam, James M Rehg, and Abhinav Gupta. No rl, no simulation: Learning to navigate without navigating. In *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, pages 26661–26673, 2021. [2](#)
- [18] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2961–2969, 2017. [2, 3, 6](#)
- [19] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16000–16009, 2022. [1](#)
- [20] Jingyang Huo, Qiang Sun, Boyan Jiang, Haitao Lin, and Yanwei Fu. Geovln: Learning geometry-enhanced visual representation with slot attention for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23212–23221, 2023. [2](#)
- [21] Minyoung Hwang, Jaeyeon Jeong, Minsoo Kim, Yoonseon Oh, and Songhwai Oh. Meta-explore: Exploratory hierarchical vision-and-language navigation using scene object spectrum grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6683–6693, 2023.
- [22] Aishwarya Kamath, Peter Anderson, Su Wang, Jing Yu Koh, Alexander Ku, Austin Waters, Yinfei Yang, Jason Baldridge, and Zarana Parekh. A new path: Scaling vision-and-language navigation with synthetic instructions and imitation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10813–10823, 2023. [2](#)

- [23] Nuri Kim, Obin Kwon, Hwiyeon Yoo, Yunho Choi, Jeongho Park, and Songhwai Oh. Topological Semantic Graph Memory for Image Goal Navigation. In *Proceedings of the Conference on Robot Learning (CoRL)*, 2022. 2
- [24] Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Matt Deitke, Kiana Ehsani, Daniel Gordon, Yuke Zhu, et al. Ai2-thor: An interactive 3d environment for visual ai. *arXiv preprint arXiv:1712.05474*, 2017. 1
- [25] Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 104–120, 2020. 2
- [26] Jacob Krantz, Stefan Lee, Jitendra Malik, Dhruv Batra, and Devendra Singh Chaplot. Instance-specific image goal navigation: Training embodied agents to find object instances. *arXiv preprint arXiv:2211.15876*, 2022. 1, 2, 5
- [27] Jacob Krantz, Shurjo Banerjee, Wang Zhu, Jason Corso, Peter Anderson, Stefan Lee, and Jesse Thomason. Iterative vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14921–14930, 2023. 2
- [28] Jacob Krantz, Theophile Gervet, Karmesh Yadav, Austin Wang, Chris Paxton, Roozbeh Mottaghi, Dhruv Batra, Jitendra Malik, Stefan Lee, and Devendra Singh Chaplot. Navigating to objects specified by images. *arXiv preprint arXiv:2304.01192*, 2023. 2, 5, 6
- [29] Obin Kwon, Jeongho Park, and Songhwai Oh. Renderable neural radiance map for visual navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9099–9108, 2023. 2
- [30] Jialu Li and Mohit Bansal. Improving vision-and-language navigation by generating future-view image semantics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10803–10812, 2023. 2
- [31] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Lightglue: Local feature matching at light speed. *arXiv preprint arXiv:2306.13643*, 2023. 4
- [32] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2023. 3, 6
- [33] Arjun Majumdar, Gunjan Aggarwal, Bhavika Devnani, Judy Hoffman, and Dhruv Batra. Zson: Zero-shot object-goal navigation using multimodal goal embeddings. In *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, pages 32340–32352, 2022. 2
- [34] Oleksandr Maksymets, Vincent Cartillier, Aaron Gokaslan, Erik Wijmans, Wojciech Galuba, Stefan Lee, and Dhruv Batra. Thda: Treasure hunt data augmentation for semantic navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15374–15383, 2021. 2
- [35] Lina Mezghan, Sainbayar Sukhbaatar, Thibaut Lavril, Oleksandr Maksymets, Dhruv Batra, Piotr Bojanowski, and Kartik Alahari. Memory-augmented reinforcement learning for image-goal navigation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3316–3323, 2022. 2
- [36] Mahdi Kazemi Moghaddam, Qi Wu, Ehsan Abbasnejad, and Javen Shi. Optimistic agent: Accurate graph-based value estimation for more successful visual navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3733–3742, 2021. 2
- [37] Arsalan Mousavian, Alexander Toshev, Marek Fišer, Jana Košecká, Ayzaan Wahid, and James Davidson. Visual representations for semantic target driven navigation. In *Proceedings of the IEEE international conference on robotics and automation (ICRA)*, pages 8846–8852, 2019. 2
- [38] Maxime Oquab, Timothée Darct, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 1
- [39] Anwesan Pal, Yiding Qiu, and Henrik Christensen. Learning hierarchical relationships for object-goal navigation. In *Proceedings of the Conference on Robot Learning (CoRL)*, pages 517–528, 2021. 2
- [40] Ruslan Partsey, Erik Wijmans, Naoki Yokoyama, Oles Dobrovych, Dhruv Batra, and Oleksandr Maksymets. Is mapping necessary for realistic pointgoal navigation? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17232–17241, 2022. 2
- [41] Santhosh K Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alex Clegg, John Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, et al. Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai. *arXiv preprint arXiv:2109.08238*, 2021. 1
- [42] Santhosh K. Ramakrishnan, Devendra Singh Chaplot, Ziad Al-Halah, Jitendra Malik, and Kristen Grauman. Poni: Potential functions for objectgoal navigation with interaction-free learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2
- [43] Ram Ramrakhya, Eric Undersander, Dhruv Batra, and Abhishek Das. Habitat-web: Learning embodied object-search strategies from human demonstrations at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5173–5183, 2022. 2
- [44] Nikolay Savinov, Alexey Dosovitskiy, and Vladlen Koltun. Semi-parametric topological memory for navigation. *arXiv preprint arXiv:1803.00653*, 2018. 2
- [45] Manolis Savva, Angel X Chang, Alexey Dosovitskiy, Thomas Funkhouser, and Vladlen Koltun. Minos: Multi-modal indoor simulator for navigation in complex environments. *arXiv preprint arXiv:1712.03931*, 2017. 2
- [46] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9339–9347, 2019. 1

- [47] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 1, 6
- [48] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 1
- [49] Andrew Szot, Alexander Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Singh Chaplot, Oleksandr Maksymets, et al. Habitat 2.0: Training home assistants to rearrange their habitat. *Advances in Neural Information Processing Systems (NIPS)*, 34:251–266, 2021. 1, 5
- [50] Michał Tyszkiewicz, Pascal Fua, and Eduard Trulls. Disk: Learning local features with policy gradient. In *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, pages 14254–14265, 2020. 4
- [51] W Wang, J Dai, Z Chen, Z Huang, Z Li, X Zhu, X Hu, T Lu, L Lu, H Li, et al. Interimage: Exploring large-scale vision foundation models with deformable convolutions. *arxiv* 2022. *arXiv preprint arXiv:2211.05778*, 2023. 2, 5, 6, 7
- [52] Justin Wasserman, Karmesh Yadav, Girish Chowdhary, Abhinav Gupta, and Unnat Jain. Last-mile embodied visual navigation. In *Proceedings of the Conference on Robot Learning (CoRL)*, 2022. 2
- [53] Fei Xia, Amir R Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson env: Real-world perception for embodied agents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9068–9079, 2018. 1, 6
- [54] Karmesh Yadav, Ram Ramrakhy, Santhosh Kumar Ramakrishnan, Theo Gervet, John Turner, Aaron Gokaslan, Noah Maestre, Angel Xuan Chang, Dhruv Batra, Manolis Savva, et al. Habitat-matterport 3d semantics dataset. *arXiv preprint arXiv:2210.05633*, 2022. 5
- [55] Karmesh Yadav, Arjun Majumdar, Ram Ramrakhy, Naoki Yokoyama, Alexei Baevski, Zsolt Kira, Oleksandr Maksymets, and Dhruv Batra. Ovrl-v2: A simple state-of-art baseline for imagenav and objectnav. *arXiv preprint arXiv:2303.07798*, 2023. 2, 5, 6
- [56] Karmesh Yadav, Ram Ramrakhy, Arjun Majumdar, Vincent-Pierre Berges, Sachit Kuhar, Dhruv Batra, Alexei Baevski, and Oleksandr Maksymets. Offline visual representation learning for embodied navigation. In *Proceedings of the Workshop on Reincarnating Reinforcement Learning at ICLR 2023*, 2023. 2
- [57] Karmesh Yadav, Ram Ramrakhy, Santhosh Kumar Ramakrishnan, Theo Gervet, John Turner, Aaron Gokaslan, Noah Maestre, Angel Xuan Chang, Dhruv Batra, Manolis Savva, et al. Habitat-matterport 3d semantics dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4927–4936, 2023. 1, 3, 6
- [58] Wei Yang, Xiaolong Wang, Ali Farhadi, Abhinav Gupta, and Roozbeh Mottaghi. Visual semantic navigation using scene priors. *arXiv preprint arXiv:1810.06543*, 2018. 2
- [59] Zijiao Yang, Arjun Majumdar, and Stefan Lee. Behavioral analysis of vision-and-language navigation agents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2574–2582, 2023. 2
- [60] Joel Ye, Dhruv Batra, Abhishek Das, and Erik Wijmans. Auxiliary tasks and exploration enable objectgoal navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16117–16126, 2021. 2
- [61] Jiazhao Zhang, Liu Dai, Fanpeng Meng, Qingnan Fan, Xuelin Chen, Kai Xu, and He Wang. 3d-aware object goal navigation via simultaneous exploration and identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6672–6682, 2023. 2
- [62] Sixian Zhang, Xinhang Song, Yubing Bai, Weijie Li, Yakui Chu, and Shuqiang Jiang. Hierarchical object-to-zone graph for object navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15130–15140, 2021. 2
- [63] Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J Lim, Abhinav Gupta, Li Fei-Fei, and Ali Farhadi. Target-driven visual navigation in indoor scenes using deep reinforcement learning. In *Proceedings of the IEEE international conference on robotics and automation (ICRA)*, pages 3357–3364, 2017. 2