IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 13 August 2025, accepted 29 August 2025, date of publication 9 September 2025, date of current version 22 September 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3607672

RESEARCH ARTICLE

Semantic Object Navigation With Segmenting Decision Transformer

ALEKSEI STAROVEROV[1], TATIANA ZEMSKOVA[1,2], DMITRY A. YUDIN[1,2], AND ALEKSANDR I. PANOV[1,2], (Member, IEEE)

[1]AIRI, 121170 Moscow, Russia
[2]Moscow Institute of Physics and Technology, 141701 Dolgoprudny, Russia

Corresponding author: Aleksei Staroverov (staroverov@airi.net)

This work was supported in part by the Ministry of Economic Development of the Russian Federation through MIPT under Grant 139-15-2025-013, and in part by IGK under Grant 000000C313925P4B0002.

ABSTRACT Object navigation remains a fundamental challenge in robotics, particularly when agents must reach targets specified by semantic categories. While existing approaches often treat semantic understanding and navigation as separate components, we demonstrate that their tight coupling is crucial for robust performance. We present SegDT (Segmenting Decision Transformer), a novel architecture that jointly learns to predict semantic segmentation masks and navigation actions through a unified transformer-based model. Our key insight is that temporal information from sequential observations can simultaneously enhance both segmentation quality and navigation decisions. To address the inherent challenges of transformer-based navigation—notably poor sample efficiency and computational complexity—we introduce a two-phase training approach: offline pretraining on expert demonstrations followed by online policy refinement through knowledge transfer from a recurrent neural network. Extensive experiments in the Habitat simulator demonstrate that SegDT achieves higher results using predicted segmentation masks, outperforming a single-frame baseline with a pre-trained semantic segmentation model and approaching the performance of systems using ground truth semantic information. Our ablation studies reveal that SegDT's temporal processing also improves segmentation quality, highlighting the synergistic benefits of joint optimization. When integrated into complete object navigation systems, SegDT enhances overall performance by 9.6% in path efficiency compared to the state-of-the-art method. The code of SegDT is made publicly available at https://github.com/CognitiveAISystems/SegDT

INDEX TERMS Navigation, reinforcement learning, semantic segmentation, robotics.

I. INTRODUCTION

Object-centric navigation for intelligent agents (e.g., robots) in unknown environments remains a significant challenge in robotics and computer vision. This is evidenced by performance metrics in modern simulation environments such as Habitat [1], AI2Thor [2], and similar platforms. Two fundamental limitations persist. First, state-of-the-art neural networks operating in real-time still struggle with reliable object segmentation, particularly for distant or partially occluded objects [3], [4]. Second, the prediction of agent actions from visual data exhibits substantial error rates across both modular approaches [5] and end-to-end neural architectures [6], indicating significant room for improvement in navigation performance.

The challenge of image sequence segmentation for intelligent agents presents additional complexities. Current approaches employ various strategies: direct fusion of image sequence features [7], [8], auto-regressive prediction of segmentation masks [9], [10], and incorporation of three-dimensional constraints [11], [12], [13], [14], including Gaussian blending techniques [15], [16]. However, these methods still demonstrate significant limitations when applied to indoor navigation scenarios, particularly in maintaining consistent object recognition across varying viewpoints and distances.

The associate editor coordinating the review of this manuscript and approving it for publication was Tai Fei.

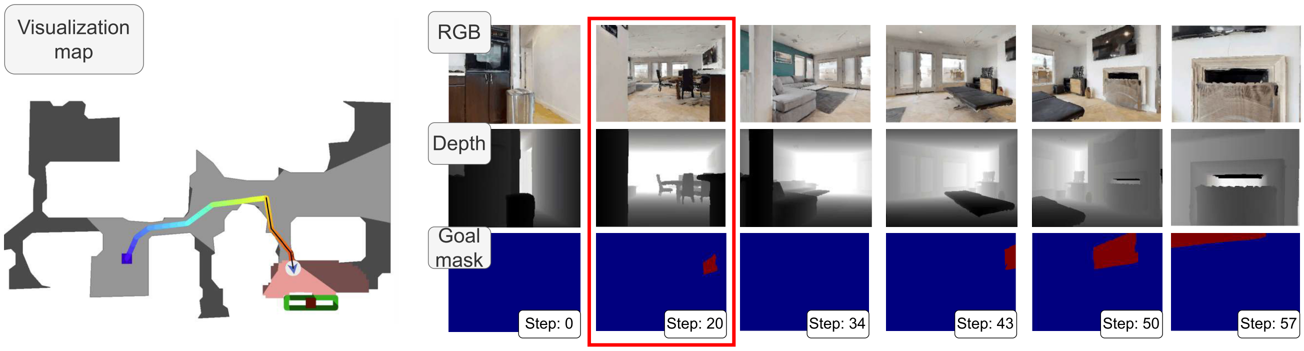


FIGURE 1. Semantic Goal Reacher task in the Habitat environment. The episode starts from the agent's initial position (left), where the target object is visible, and the objective is to reach within 1.0 m of the target (right). In a full ObjectNav pipeline, earlier steps of locating the target can be performed by exploration algorithms; GoalReacher is the critical final stage that efficiently approaches the goal once it is detected. On the visualization map, the black segment of the trajectory corresponds to the GoalReacher phase, and the red zone indicates the success region where the episode is considered successful.

Navigation tasks inherently represent partially observable reinforcement learning (RL) problems where historical context must be processed by sequence models [17]. While transformers have demonstrated remarkable capabilities in computer vision and natural language processing tasks [18], [19] and exhibit strong long-term memory capabilities [20], they generally suffer from poor sampling efficiency and do not improve long-term credit assignment compared to recurrent neural networks (RNNs) [21]. To address these limitations, we propose a novel method that simultaneously trains RNN-based and transformer-based policy versions. This approach leverages the RNN-based policy's ability to effectively solve navigation tasks using ground truth segmentation from the simulator, while enabling the transformer-based policy to predict segmentation from RGB frame sequences and generate action sequences through knowledge transfer from the RNN-based policy.

These challenges are particularly evident in Semantic Object Navigation tasks which, in the literature [22], are defined as scenarios where an agent, randomly initialized within an unfamiliar environment, must navigate toward an instance of a specified object category $C \in \{c_1, c_2, \ldots, c_n\}$ (e.g., a *plant*). Typically, modular methods [5], [23], [24] decompose ObjectNav into sub-skills including exploration, recognition, and goal reaching, with specialized components for each sub-skill. The solution generally consists of two stages: environment exploration to locate an instance of the given semantic goal, followed by reaching the identified object. The latter stage presents particular challenges in unseen scenes and object episodes, as it heavily relies on semantic goal understanding. Recent work [25] demonstrates that incorporating bee-lining (goal reaching) capabilities with pretrained object detection networks into existing end-to-end solutions can significantly improve overall ObjectNav metrics on validation episodes.

In this work, we propose to unify action prediction and RGB-D image sequence segmentation within a single transformer model, focusing specifically on the semantic goal-reaching stage. We initialize the agent at a random viewpoint of the semantic goal at a maximum distance of ten meters (Fig. 1). Our results demonstrate that this unified approach enhances both image segmentation quality and action generation accuracy in solving the navigation problem for semantically-specified objects. This semantic object navigation capability has practical applications in robotics scenarios where embodied agents must navigate in non-deterministic environments [22].

The main contributions of this work include:

- We introduce SegDT, a novel multimodal transformer architecture that jointly learns segmentation and navigation policies by processing sequences of RGB-D frames. Our model leverages temporal information to simultaneously improve segmentation quality and navigation performance through a unified training objective, unlike previous approaches that treat segmentation only as an auxiliary task.
- We develop a two-phase training strategy that addresses fundamental limitations of transformer-based navigation: poor sampling efficiency and inadequate long-term credit assignment. Our approach combines offline pretraining on collected trajectories with online policy fine-tuning through a knowledge transfer mechanism between an RNN-based policy (using ground-truth segmentation) and our transformer-based model (which learns to predict segmentation).
- Through extensive experiments in Habitat Sim, we demonstrate that our unified approach achieves superior performance compared to both traditional navigation methods and state-of-the-art approaches. Notably, SegDT maintains robust navigation performance when transitioning from ground-truth to predicted segmentation masks and improves segmentation quality by aggregating temporal information during navigation.

The code of SegDT is made publicly available at https://github.com/CognitiveAISystems/SegDT.

II. RELATED WORK

Recent methods for object goal navigation use scene semantic information for action prediction to reduce overfitting and increase the navigation quality for unseen environments. The scene semantic can be available in the form of a 2D semantic segmentation mask. For instance, authors of the THDA method [26] introduce a policy network that uses depth and multichannel semantic masks as input. SkillFusion approach [27] proposes a goal-reaching policy that leverages an RGB observation and a binary segmentation mask of the object goal. During inference time, the success rate of such navigation approaches heavily relies on the quality of input segmentation masks [27]. Despite the active development of neural network architectures, the state-of-the-art methods for semantic segmentation (e.g. Mask2Former [28], OneFormer [29], OpenSeeD [30], MQ-Former [31]) still show imperfect segmentation quality, especially for indoor environments, where objects can vary a lot within one semantic category.

In addition, the state-of-the-art methods for semantic segmentation do not take into account the peculiarities of an embodied agent interacting with its environment during navigation. The agent has a limited field of view; therefore, instant observations may contain erroneous semantics when looking at the object from certain view angles. During the navigation episode, the agent can update its semantic understanding of the scene by observing the scene from more advantageous viewpoints. Such refinement can occur explicitly by using the accumulated semantic map of the environment [32], [33], [34]. The explicit semantic maps of the environment can be used as input to predict action policy [11], [23], [35]. Other methods, such as [6], use implicit maps to model the history of observations. However, to build a semantic map, one needs to have information about the agent's pose at each moment in time, while our method uses only RGB images and depth maps as input.

We use a method that aggregates sequence information from previous semantic observations to refine semantic segmentation on the current frame and predict the next action. In this sense, our method is related to methods that solve the task of video segmentation [36], [37] and Vision-Language-Navigation models such as NavGPT [38]. However, unlike such methods, our approach allows the agent to control its observations to navigate to the goal and improve the segmentation quality. At the same time, our method differs from existing embodied computer vision methods [39], [40], [41], [42], [43], [44]. These methods aim to improve the quality of visual perception, while our method increases both the quality of navigation and the quality of segmentation. The methods for embodied computer vision often operate in the next-best-view paradigm or use a small sequence of frames to predict the next action. However, the agent needs a longer history of observations to successfully solve the object goal navigation task. Unlike [7], we consider a complex photo-realistic 3D environment of the HM3DSem v0.2 [45] scenes.

A special feature of our method is the joint training of a semantic segmentation model and a transformer to predict the next actions. Our work aligns with transformer-based segmentation advances in challenging environments [46], [47], [48]. Additionally, transformer-based scene segmentation with unsupervised domain adaptation has been explored in [49]. Previous works [26], [50] consider semantic loss as an additional task for model training. However, these methods use semantic loss only to improve the action policy, and not to improve the quality of semantic segmentation by aggregating information from a sequence of frames.

III. TASK SETUP

We formulate the Semantic Goal Reaching task as a Partially Observable Markov Decision Process (POMDP) defined by the tuple $(S, A, P, R, \rho_0, \gamma)$ for the underlying observation space S, action space A, transition distribution P, reward function R, initial state distribution ρ_0, and discount factor γ.

At each timestep t, the agent receives an observation o_t consisting of:

- An RGB-D image pair (I_t, D_t), where $I_t \in \mathbb{R}^{H\times W\times 3}$ and $D_t \in \mathbb{R}^{H\times W}$.
- A target category $c \in \mathcal{C}$, where $\mathcal{C}$ is the set of valid object categories.

The action space $\mathcal{A}$ consists of six discrete actions: `callstop` to terminate the episode, `forward` by 0.25 m, `turnleft` or `turnright` by angle 15°, `lookup`, `lookdown` by turning the agent's head by angle 30°. Episodes are initialized with the agent randomly placed at a position where the target object is within the agent's field of view and the geodesic distance to the target is at most 10 meters. An episode terminates when either the agent executes the `callstop` action or the maximum episode length of 64 timesteps is reached.

Following standard metrics in object navigation literature [22], we evaluate performance using:

- Success Rate (SR): Percentage of episodes where the agent successfully reaches within 1.0m of the target.
- Success weighted by inverse Path Length (SPL): SR weighted by the ratio of shortest path to actual path length.
- SoftSPL: A soft version of SPL that provides partial credit for incomplete episodes.

This formulation builds upon prior work in visual navigation [22] while specifically focusing on the goal-reaching phase, where semantic understanding of the target object is crucial for success.

A. REWARD FUNCTION

During online fine-tuning, we use a shaped per-step reward together with a terminal bonus. Let d_t be the geodesic distance from the agent to the closest valid point of the target object at time t, and let s_t denote the area (in pixels) of the predicted binary mask of the target object in the current

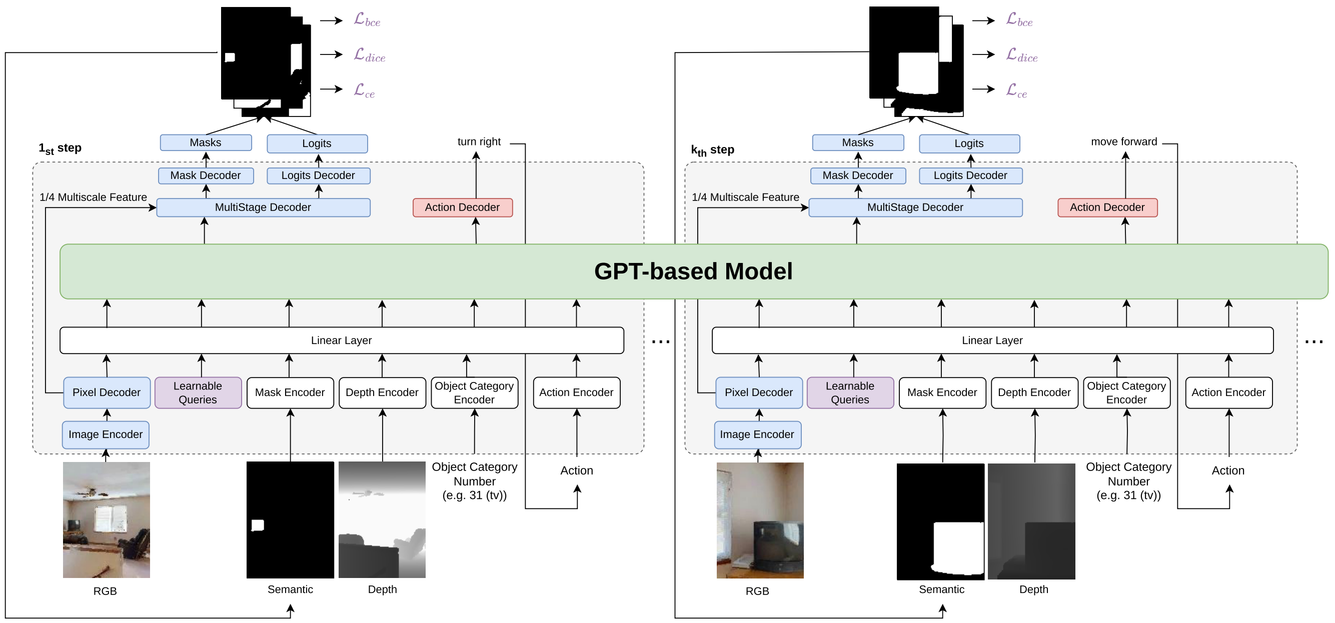


FIGURE 2. **Architectural overview of the proposed Segmenting Decision Transformer (SegDT). The model processes multimodal inputs through parallel streams: (1) a segmentation branch using an image encoder, pixel decoder, and multistage mask decoder, and (2) a decision branch that aggregates temporal information to predict actions. The transformer architecture enables joint optimization of both segmentation quality and navigation policy through shared representations.**

frame. The per-step reward is

$$r_t = -0.03 + \text{clip}(d_{t-1} - d_t,\ -1,\ 1) + \text{clip}(s_t - s_{t-1},\ -1,\ 1),$$

where the first term is a small-time penalty, the second term rewards progress toward the goal, and the third term rewards increased evidence of the goal in the current observation via the predicted mask.

When the agent executes the `callstop` action, a terminal reward is given

$$r_T = \begin{cases} 10, & \text{if } d_t \le 1.0\,\text{m and } s_t \ge \tau, \\ 5, & \text{if } d_t \le 1.0\,\text{m and } s_t < \tau, \\ -1, & \text{otherwise,} \end{cases}$$

with threshold $\tau = 100$ pixels.

IV. METHOD

We modify the original Decision Transformer architecture [51] from two perspectives. First, we add a segmentation branch, demonstrating the synergy between semantic segmentation and goal reaching tasks. Second, we propose an adaptation of the Decision Transformer training method, which enables SegDT to be effectively fine-tuned in an online environment, thereby improving navigation performance compared to training on offline data. Prediction at time t involves two stages. An observation at time t consists of an image I_t, a depth map D_t, and a target category name c. First, multi-scale feature maps of I_t are generated using an image encoder and a pixel decoder. These feature maps, along with trainable query features, are then fed into the decision transformer. After processing, the trainable query features are decoded by a multi-stage mask decoder to generate segmentation masks for a fixed set of categories. From the set of masks, a binary mask for the target category is selected, and its embedding is extracted. This embedding, combined with the depth map and the category name embedding, completes the observation sequence embeddings. In the second step, the full sequence of observation embeddings is fed into the decision transformer to predict the probability distribution and state value of the next action. We then sample action a_t and add its embedding to the observation sequence to predict actions at time $t + 1$. Figure 2 illustrates the model architecture.

A. SEGMENTATION MODULES

When choosing the architecture of the Segmenting Decision Transformer (SegDT) modules responsible for segmentation, we take Mask2Former [28] as a basis. Mask2Former is one of the state methods for semantic segmentation. This method considers the segmentation problem as a problem of predicting a set of binary masks and their classification. The segmentation model is given an image of size (H, W, C) as input.

The main components of Mask2Former are an image encoder, a pixel decoder, and a multistage decoder. We use ResNet50 as the backbone. The output of the backbone is fed to the pixel decoder to generate 4 maps of high-resolution per-pixel embeddings. The per-pixel embeddings have $1/4$, $1/8$, $1/16$, and $1/32$ of the resolution of the input image. We use a $1/32$ per-pixel embedding map as the image embedding for the Transformer model input.

In the original single-frame Mask2Former model, binary segmentation masks and their classification logits are decoded from N learnable query features using multiscale feature maps. In our work, we use N learnable query features as input to the Transformer model to take into account the context of previous observations. After passing through the transformer, the updated query features are passed through the multistage decoder. Here, similar to the Mask2Former model, we use multi-scale feature maps to predict binary segmentation masks and their logits. From these binary masks, a multi-channel semantic segmentation mask is formed for $N_{cl} = 40$. We then select the target semantic mask and use the ResNet50 encoder to create a semantic feature of size $(1, d_{sem})$. This feature describes the presence of the semantic goal in the current observation, similar to TDHA [26].

1) OBSERVATIONS EMBEDDINGS

For each time point, we describe the current observation using 29 embeddings obtained from different encoders and projected into the GPT hidden dimension $d_{GPT} = 768$. For each of the T-frames, we flatten the image pixel embeddings from Mask2Former into a sequence and project the image embeddings into d_{GPT} using a linear layer. Thus, the image embedding for an image has a dimension of $(H \cdot W/32, d_{GPT})$. The learnable queries are represented by a set of 50 embeddings with dimension $(1, d_{GPT})$. We encode the semantics of each image using ResNet50 features obtained from the binary segmentation mask of the target object into a feature vector of dimension $(1, d_{sem})$. Thus, after projection, the embedding of semantic predictions for 1 image has a dimension of $(1, d_{GPT})$. We encode depth for each of the observations using ResNet18, resulting in a feature vector of dimension $(1, d_{depth})$. Using a linear layer, we project the depth features into the d_{GPT} feature space. Thus, the feature embedding of the depth observation has dimension $(1, d_{GPT})$. To encode the target category and the performed action, we use a look-up table of learnable embeddings of dimensions (N_{cl}, d_{GPT}) and $(N_{actions}, d_{GPT})$, respectively. We populate the GPT input sequence with T observation embeddings. Thus, the dimension of the input sequence of observation embeddings is $(T \cdot (H \cdot W/32 + 4), d_{GPT})$.

2) PREDICTIONS

Since the goal of the semantic object navigation task is to reach an object of a certain target category, we expect that using the observation history can improve the segmentation quality for this target category. To decode semantic predictions, we use an idea from the original Mask2Former segmentation model [28]. We take the output learnable query features from the SegDT and pass them through the multistage decoder. To obtain the binary segmentation masks and their logits at time t, we additionally use the multi-scale feature maps predicted by the pixel decoder at time t. We use MLPs to decode the action distribution for the actor head.

To predict the action at step t, we use the set of observations $\{o_0, \ldots, o_t\}$ and the previous actions $\{a_0, \ldots, a_{t-1}\}$. First, the sequence $\{o_0, a_0, \ldots, o_{t-1}, a_{t-1}, o_t\}$ is passed to the SegDT input to predict the segmentation masks $\{M_i^{pred}\}_{i=0}^{t}$. The mask corresponding to the target object category is used as the semantic observation for the time t. Next, SegDT makes another prediction of the action a_t, taking into account the segmentation mask, the depth, and the target category at time t. In this case, the last token of the output sequence of the transformer is used as input of the action decoder, i.e., the last token of the observation o_t.

B. LEARNING PROCESS

1) JOINT LEARNING ON OFFLINE DATA

As a central aspect of our experiment, we initialize the image encoder, the pixel decoder, and the multi-stage decoder responsible for segmentation prediction with parameters of a pre-trained segmentation model. The primary goal during the initial phase of training is to establish an effective representation of the observations intended for navigation. To achieve this goal, we rely on an offline demonstration dataset composed of semantic goal-reaching instances between the start coordinates and the most proximal target. We collect the action probability distribution of a pre-trained RL agent with RNN and ground truth segmentation as input. During these initial stages, both SegDT (our multi-stage mask decoder) and our action decoder are trained simultaneously. To optimize mask prediction, we use the sum of the pixel-by-pixel binary cross-entropy $\mathcal{L}_{bce}$, the dice loss $\mathcal{L}_{dice}$, and the cross-entropy loss $\mathcal{L}_{ce}$ for mask classification as our loss function. Behavior cloning $(\mathcal{L}_{bce})$ is used to predict the action sequence.

$$\mathcal{L}_{total} = \lambda_{segm}\mathcal{L}_{segm} + \lambda_{bce}^{act}\mathcal{L}_{bce}, \tag{1}$$

$$\mathcal{L}_{segm} = \lambda_{bce}^{segm}\mathcal{L}_{bce} + \lambda_{dice}\mathcal{L}_{dice} + \lambda_{ce}\mathcal{L}_{ce}. \tag{2}$$

We use the following values of hyper-parameters: $\lambda_{segm} = 1$, $\lambda_{bce}^{act} = 1$, $\lambda_{MSE} = 0.1$, $\lambda_{bce}^{segm} = 5$, $\lambda_{dice} = 5$, $\lambda_{ce} = 2$.

2) ONLINE FINETUNING

While behavior cloning provides a strong initialization for our policy, it suffers from two fundamental limitations: distribution shift between training and deployment states, and an inability to improve beyond the demonstrator's performance. To address these limitations, we employ online reinforcement learning to continuously adapt the policy through direct environment interaction.

However, online reinforcement learning (RL) requires a significant number of samples to achieve robust performance, which can be a significant limitation. Additionally, the use of the transformer model introduces substantial computational cost, especially for long sequences, as causal transformers require $O(t^2)$ time to compute the representation at time step t, so they typically exhibit poor sample efficiency compared to recurrent architectures (10 times more during

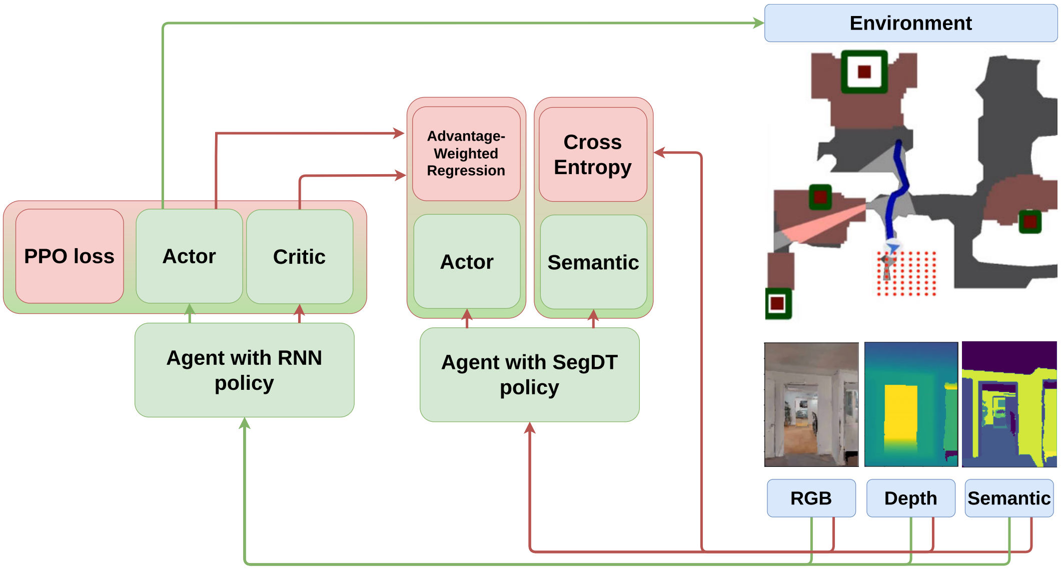


FIGURE 3. Schematic representation of the online fine-tuning process for SegDT. The RNN-based policy, operating with ground-truth segmentation, generates expert trajectories that guide the transformer through advantage-weighted regression loss. This knowledge transfer mechanism enables efficient learning of both segmentation and navigation capabilities while maintaining computational tractability.

our experiments), requiring more training time to achieve comparable performance.

To overcome these challenges, we introduce a novel knowledge transfer approach, shown in Figure 3. Let $\pi_{\text{RNN}}(a_t|s_t)$ denote an RNN-based policy trained using ground truth segmentation masks, and $\pi_{\text{SegDT}}(a_t|s_t)$ represent our transformer-based policy that must learn to predict segmentation masks. We simultaneously train both policies using the following composite loss for π_{SegDT} and π_{RNN}:

$$\mathcal{L}_{\text{total}} = \mathcal{L}^{\pi_{\text{RNN}}}_{\text{PPO}} + \lambda_{\text{AWR}}\mathcal{L}^{\pi_{\text{SegDT}}}_{\text{AWR}} + \lambda_{\text{segm}}\mathcal{L}^{\pi_{\text{SegDT}}}_{\text{segm}}, \tag{3}$$

$$\mathcal{L}^{\pi_{\text{SegDT}}}_{\text{AWR}} = \sum_{i=0}^{n-1} \log \pi_{\text{SegDT}}(a_{t+i}|s_t, a_{t:t+i-1}) * \exp(A^{\pi_{\text{RNN}}}_{t-1}/\beta), \tag{4}$$

where $\mathcal{L}_{\text{PPO}}$ is the standard PPO objective that is used for the $\pi_{\text{RNN}}(a_t|s_t)$ policy, $\mathcal{L}_{\text{AWR}}$ [52] encourages the transformer policy to match the RNN policy's action distributions considering the advantage calculated by the $\pi_{\text{RNN}}(a_t|s_t)$ policy, and $\mathcal{L}_{\text{seg}}$ is the segmentation loss defined in Equation 2. The hyperparameters we used: $\lambda_{AWR} = 2$, $\lambda_{segm} = 1$, $\beta = 1$.

This approach leverages the sample efficiency of the RNN policy while allowing the transformer to simultaneously improve its segmentation predictions and action selection. The RNN policy, with access to ground truth segmentation, can quickly learn effective navigation strategies. Through knowledge distillation, these strategies are transferred to the transformer policy, which must additionally learn to predict accurate segmentation masks from raw observations.

Our empirical results demonstrate that this training strategy enables SegDT to achieve performance comparable to the RNN policy with ground truth segmentation, despite operating solely with predicted segmentation masks during deployment. This suggests that the joint optimization of navigation and segmentation allows the model to develop robust representations that support both tasks.

V. EXPERIMENTS

The main goal of our experiments is to navigate an autonomous agent toward its target object by minimizing cumulative distance and maximizing the understanding of the environment. To achieve this, we have followed a twofold training phase strategy: with the first phase focusing on obtaining high-quality semantic segmentation masks, and the second phase shifting towards action prediction with the use of an online Reinforcement Learning method for an adaptable learning experience.

A. EXPERIMENTAL SETUP

1) DATASETS

The experiments were carried out in the Habitat environment [1]. For the experiments, we selected 146 training and 36 validation scenes of the HM3DSem v0.2 dataset [45]. These scenes were divided into a training set of 173 scenes and a validation set of 9 scenes. We provide a visualization of the selected scenes in Figure 4. These validation scenes were not included in the training set for either the Mask2Former segmentation model (including the pretraining phase) or SegDT. Thus, our validation experiments are conducted in unseen environments that contain various room types: living room, kitchen, bedroom, bathroom, and office. Next, we sample episodes in each scene. The episode is characterized by the agent's starting position, the coordinates, and the semantic

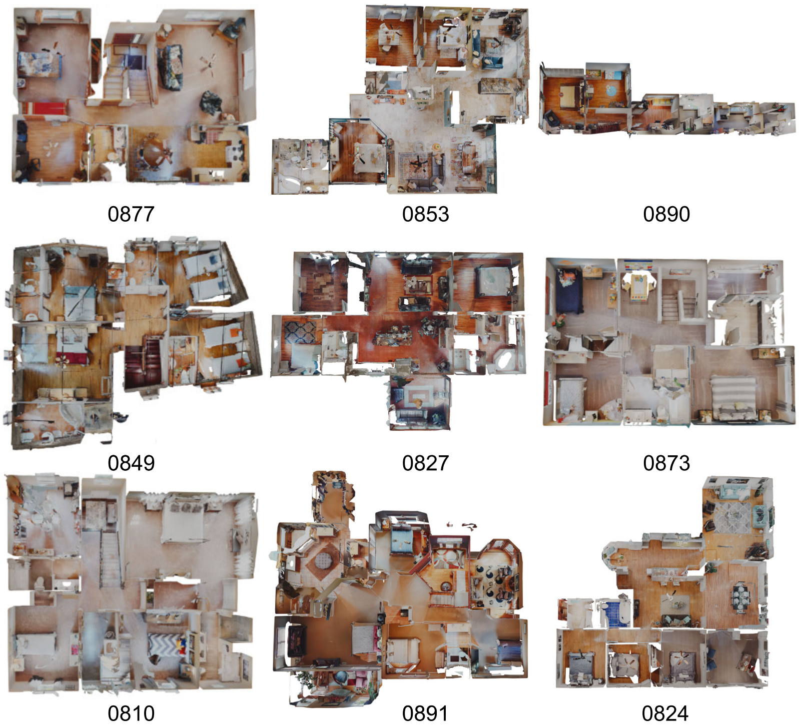


FIGURE 4. Visualization of scenes from the HM3DSem validation subset used for validation experiments. We use the following validation scenes: 0877-4ok3usBNeis, 0853-5cdEh9F2hJL, 0890-6s7QHgap2fW, 0849-a8BtkwhxdRV, 0827-BAbdmeyTvMZ, 0873-bxsVRursffK, 0810-CrMo8WxCyVb, 0891-cvZr5TUy5C5, 0824-Dd4bFSTQ8gi.

type of the target object. We randomly sample starting points for episodes satisfying two conditions of the Goal Reaching task: the target object is in the agent's field of view, and the agent is no more than 10 meters away from the goal.

For offline training of SegDT, we collect a dataset consisting of 16080 episodes in our 173 training scenes. The ground truth trajectories for behavioral cloning were obtained from the state-of-the-art RL algorithm for object goal navigation [27] using ground truth segmentation as input. The dataset for offline training contains 40 categories of the Matterport3D dataset [53] as goals for navigation, except 12 object categories: curtain, ceiling, column, door, floor, misc, objects, stairs, unlabeled, wall, window, and picture.

2) OFFLINE TRAINING

We pre-train the Mask2Former segmentation model on a dataset consisting of 125K images collected in HM3DSem v0.2 training scenes with the same training parameters as in the original Mask2Former paper [28].

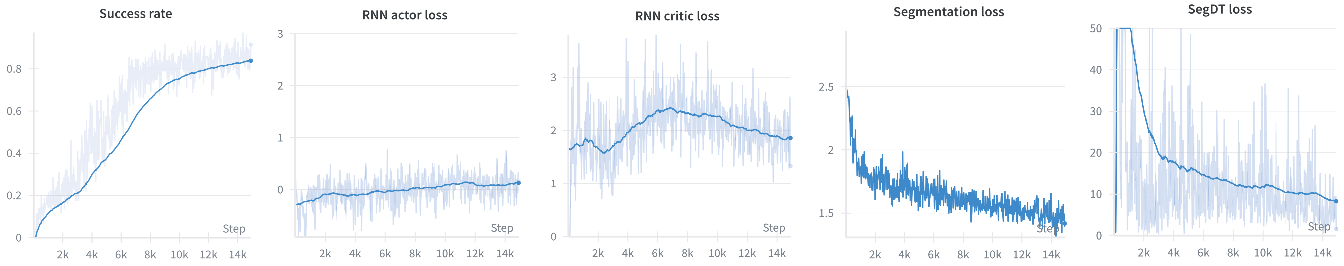
For training and validation of SegDT, we use the rendering parameters from the Habitat Challenge 2023 [54], except for the image resolution, which is changed to a lower 160 × 120 to save computational resources. As input to Mask2Former, we use the square input resolution, i.e., pad images to 160 × 160. For baseline methods, we compare SegDT against, we use the rendering parameters provided by the authors of the methods. During offline training, we freeze Image Encoder, Pixel Decoder, and MultiStage Decoder with Mask and Logits Decoders. To train the remaining Decision Transformer module and learnable query features of SegDT, we use the AdamW [55] optimizer with a learning rate of 3×10^{-4}, $\beta_1 = 0.9$, $\beta_2 = 0.98$, $\lambda = 0.01$, and linear decay of learning rate. We use a batch size equal to 8 and a maximum of 64 frames from GT trajectories during training. The parameters of pretrained Mask2former are used to initialize the parameters of the segmentation modules of SegDT.

3) ONLINE FINE-TUNING

As an RL algorithm, we use PPO with Generalized Advantage Estimation [56]. We set the discount factor γ to 0.99 and the GAE parameter τ to 0.95. Each worker collects (up to) 64 frames of experience from 18 agents running in parallel (all in different scenes) and then performs 5 epochs of PPO. We use Adam [57] with a learning rate of 1×10^{-5}. The shaped and terminal rewards follow the definition in the Task Setup paragraph above.

B. TRAINING DYNAMICS AND STABILITY

We track the optimization signals of all components during training to ensure robust convergence: the PPO objectives for the auxiliary RNN teacher, the advantage-weighted regression (AWR) loss that guides SegDT, and the total



(a) (b) (c) (d) (e)

FIGURE 5. **Training dynamics of SegDT during online fine-tuning. Shown: (a): SR moving average, (b) and (c): RNN PPO actor/critic losses (teacher), (d): total segmentation loss and (e): SegDT AWR loss (student). Solid lines denote moving averages; shaded bands indicate variability across parallel workers. All curves evolve smoothly without spikes, evidencing stable joint optimization.**

segmentation loss $\mathcal{L}_{segm}$ (Eq. 2). Figure 5 summarizes these curves.

Across runs, we observe:

- **Monotonic performance growth:** the moving average of Success Rate increases steadily and plateaus smoothly, with narrowing variance over time.
- **Well-behaved control losses:** PPO policy/value losses of the RNN teacher decrease with only mild on-policy oscillations typical of PPO, and no signs of instability.
- **Stable knowledge transfer:** the AWR loss for SegDT decreases gradually, indicating that matching the teacher distribution remains well-conditioned.
- **Consistent perception improvement:** the total segmentation loss $\mathcal{L}_{segm}$ decreases throughout training, confirming that temporal aggregation does not conflict with policy learning.

These trends collectively demonstrate that our unified training procedure is stable: perception and action objectives optimize jointly without interference, and no regime shifts or loss spikes are observed.

C. SENSITIVITY TO EXPERT POLICY QUALITY

To assess how susceptible SegDT is to the quality of the expert RNN policy used for knowledge transfer, we perform an ablation in which the student is distilled from teacher checkpoints during online fine-tuning of both (RNN and SegDT) models (Fig. 6). All evaluations in this subsection use the 79° field-of-view configuration, and each validation episode is allowed up to 500 environment steps.

Conclusion. SegDT benefits from stronger teachers, but its performance is proportional to expert quality. Advantage-weighted regression, together with the auxiliary segmentation objective, provides a stable training signal that allows the student to learn effectively even when the teacher is only moderately competent.

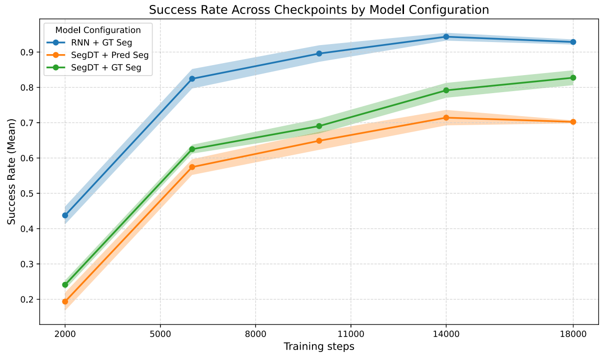


FIGURE 6. **Effect of expert policy quality on SegDT. Curves show the evolution of the success rate during online fine-tuning when distilling from weak, medium, and strong RNN teachers. Shaded regions indicate variation across seeds. The dependency is monotonic yet moderate, evidencing robustness to suboptimal experts.**

D. ONLINE SEGMENTATION QUALITY

1) BASELINE SEGMENTATION

SegDT aggregates information from several previous frames to improve the segmentation quality for the current frame. Therefore, we compare the performance of the SegDT approach with the Single Frame Mask2Former [28] baseline that makes predictions for the same frame sequence as SegDT. The Single Frame Mask2Former segments every frame in the sequence individually. We expect segmentation improvement for episodes where the agent frequently observes the target object. Therefore, we evaluate the segmentation quality on shortest path trajectories for all 112 validation episodes. The shortest path trajectories were obtained from a classical planning algorithm [58]. This planner greedily fits actions to follow the geodesic shortest path between the agent's starting point and the goal position. For each step t, we consider as a baseline segmentation the Single Frame Mask2Former masks predicted for the input image I_t.

2) SEGMENTATION METRIC

SegDT uses only target object masks to predict actions, so the navigation quality depends primarily on the quality of segmentation of these categories. For each episode, we compute the standard mean Intersection over Union ($mIoU$) [29] metric for six target categories: sofa, TV, chair, plant, toilet, and bed. We then average the resulting values across all episodes.

E. RESULTS

We evaluate our approach through comprehensive comparisons with state-of-the-art methods and detailed ablation

TABLE 1. Comprehensive performance comparison between SegDT and state-of-the-art object navigation methods on Goal Reacher task. Notable differences in sensor requirements and environmental parameters (FOV, turn angle) are included to ensure fair comparison. Results demonstrate SegDT's superior performance despite more constrained operating conditions.

Method	Sensors	FOV	Turn Angle	SR	SPL	SoftSPL
DD-PPO (500 steps) [59]	RGB-D, GPS+Compass	79°	30°	10.2	2.1	14.6
OnavRIM [6]	RGB-D, GPS+Compass	79°	30°	5.6	2.2	40.1
OnavRIM (500 steps) [6]	RGB-D, GPS+Compass	79°	30°	56.3	16.2	17.3
PIRLNav [60]	RGB-D, GPS+Compass	79°	30°	51.7	41.5	49.2
PIRLNav (500 steps) [60]	RGB-D, GPS+Compass	79°	30°	68.7	43.3	44.6
SegDT with predicted segmentation	RGB-D	79°	30°	53.4	50.1	56.7
SegDT with predicted segmentation (500 steps)	RGB-D	79°	30°	69.4	**58.8**	**58.9**
RL with RNN and GT segmentation	RGB-D	42°	15°	49.1	36.4	58.5
SegDT with GT segmentation	RGB-D	42°	15°	47.3	44.7	56.3
RL with RNN and predicted segmentation	RGB-D	42°	15°	31.2	28.2	46.2
SegDT with predicted segmentation	RGB-D	42°	15°	40.2	38.3	51.5

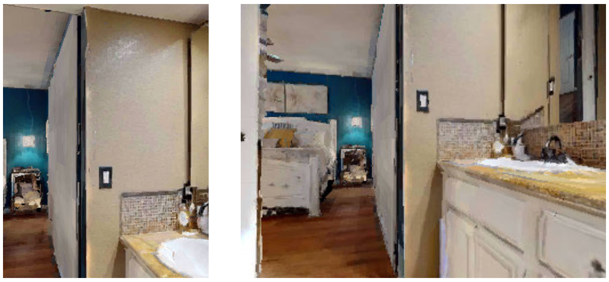
TABLE 2. Ablation study analyzing the relationship between segmentation quality and navigation performance. Results compare ground truth segmentation, single-frame Mask2Former predictions, and SegDT's temporal approach. Mean Intersection over Union (mIoU) is computed on Shortest Path Follower trajectories to provide standardized evaluation conditions. The analysis demonstrates the effectiveness of temporal information in improving both segmentation accuracy and navigation metrics.

Navigation semantics	Frame Sequence	mIoU (SPF trajectories)	SR	SPL	SoftSPL
GT	Single Frame	–	47.3	44.7	56.3
Mask2Former	Single Frame	51.8	38.0	36.2	49.9
SegDT	**Navigation**	**53.7**	**40.2**	**38.3**	**51.5**

studies. All evaluations use a validation set of 112 episodes across 9 scenes, with 6 target object categories from the Habitat Challenge.

1) COMPARISON WITH STATE-OF-THE-ART METHODS

Table 1 presents a comprehensive comparison between SegDT and existing object navigation approaches. Several key observations emerge. DD-PPO achieves only a 10.2% success rate with 2.1% SPL, highlighting the limitations of methods that don't explicitly leverage semantic understanding. Methods like OnavRIM and PIRLNav, while achieving higher success rates with extended 500-step episodes (56.3% and 68.7% respectively), show significantly lower path efficiency compared to SegDT on the Goal Reacher task. This is likely due to their exploration-heavy approach inherited from human demonstrations. Our SegDT model achieves strong performance even with more constrained operating conditions (42° FOV vs 79° FOV used by other methods) and without requiring GPS/compass sensors. With ground truth segmentation, SegDT achieves 47.3% SR and 44.7% SPL, demonstrating efficient goal-reaching behavior. We fine-tune SegDT using a wider field of view, 79°, and 30-degree rotation angles. Table 1 shows that with this sensor configuration, SegDT achieves high navigation performance, demonstrating a +15.5% SPL improvement over the SOTA method PIRLNav, while achieving a comparable SR relying solely on RGB and depth images. Changing the camera's field of view does not alter the model architecture. However, a wider field of view facilitates the navigation task, as a single observation contains more information about the surrounding environment (see Figure 7). In particular, a single frame contains more objects, and they do not disappear when the view is rotated.


FOV 42

FOV 79

FIGURE 7. Comparison of RGB observations with rendering parameters of 42 and 79 FOV. Visually, it can be seen that for navigation, the wider 79 FOV is an easier setting, as it provides more information within a single observation.

2) IMPACT OF SEGMENTATION QUALITY

Table 2 analyzes how different segmentation approaches affect navigation performance. Navigation with ground truth segmentation provides the upper bound performance (47.3% SR, 44.7% SPL). Single-frame Mask2Former predictions achieve 51.8% mIoU on shortest path trajectories, resulting in 38.0% SR and 36.2% SPL. SegDT's temporal approach improves segmentation quality to 53.7% mIoU, leading to better navigation performance (40.2% SR, 38.3% SPL). This demonstrates the value of incorporating temporal information for both segmentation and navigation.

We investigate how the goal mIoU metric is distributed for each category and how it relates to the successful

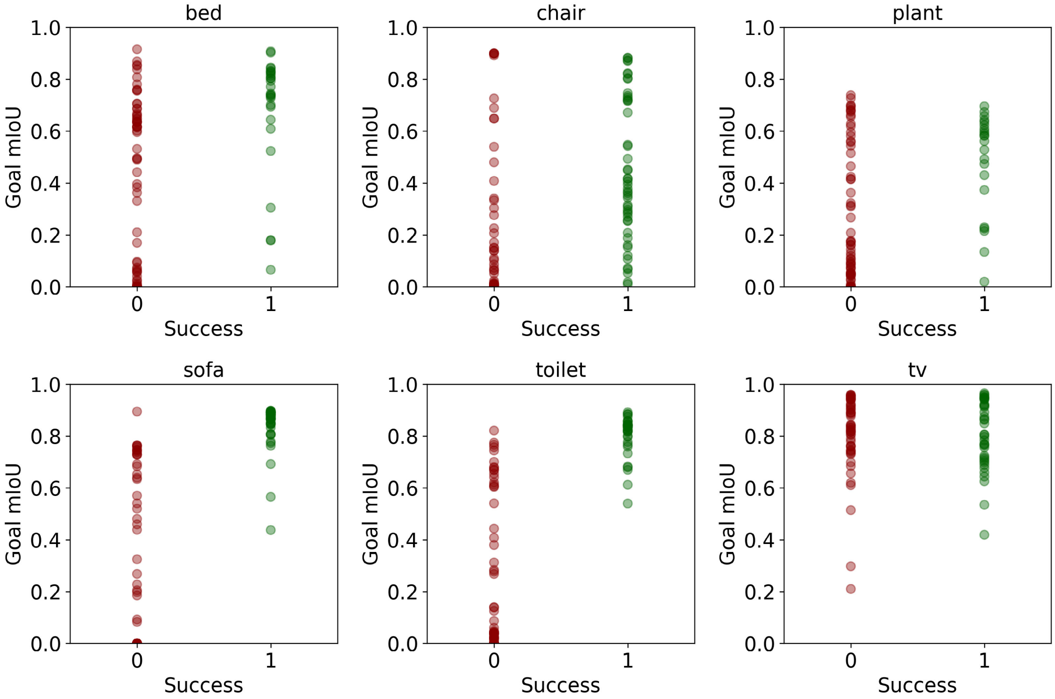


FIGURE 8. Per-category analysis of the relationship between Goal Reaching episode success and the goal mIoU metric for the corresponding category. For most categories (bed, plant, sofa, toilet), we observe a correlation between successful episode completion and relatively high goal mIoU for the corresponding target object. Episodes with high goal mIoU but unsuccessful completion correspond to errors related to navigation.

or unsuccessful completion of episodes containing that category. For the binary target mask in each episode of length N, we define goal mIoU as follows:

$$\text{Goal mIoU} = \frac{\sum_{t=1}^{T} \text{Intersection}(M_t^{\text{pred}}, M_t^{\text{gt}})}{\sum_{t=1}^{T} \text{Union}(M_t^{\text{pred}}, M_t^{\text{gt}})}, \quad (5)$$

where M_t^{pred} is the predicted target mask, M_t^{gt} is the ground-truth target mask, $t \in [1, \ldots, T]$ are the timesteps of the episode where either M_t^{pred} or M_t^{gt} is present, $\text{Intersection}(M_t^{\text{pred}}, M_t^{\text{gt}})$ is the number of pixels in the intersection of M_t^{pred} and M_t^{gt}, and $\text{Union}(M_t^{\text{pred}}, M_t^{\text{gt}})$ is the number of pixels in their union.

Figure 8 shows the results of the per-category analysis. For most categories (*bed*, *plant*, *sofa*, *toilet*), there is a clear trend that successfully completed episodes tend to have a relatively high goal mIoU. The share of unsuccessful episodes with high goal mIoU corresponds to episodes with navigation errors. Objects in the *tv* category are recognized well in both successful and failed episodes, indicating that the main difficulty for this category lies in navigation.

The distribution modes where successful completion occurs despite relatively low goal mIoU for categories such as *chair* and *plant* may correspond to objects with complex shapes. In this case, the object is correctly recognized, but the precise shape of the mask contains errors. These episodes may also correspond to navigation success achieved due to RGBD modalities.

TABLE 3. Analysis of the Role of the Segmentation Branch for the SegDT Model. The presence of a target binary mask encoder and learnable queries for segmentation in the SegDT architecture improves the adaptation quality of the strategy to new scenes in the validation set.

Method	Target mask	SR	SPL	SoftSPL
DT	no	33.4	29.7	45.9
SegDT	**yes**	**40.2**	**38.3**	**51.5**

3) ROLE OF THE SEGMENTATION BRANCH FOR THE SEGDT MODEL

We perform an ablation study to assess how the presence of a segmentation branch affects navigation quality in the SegDT architecture. To this end, we conduct an experiment using our own implementation of the Decision Transformer: a version of SegDT in which the encoded target binary mask and learnable queries are removed from the observation embeddings. This results in a transformer-based baseline for our method, with the segmentation branch entirely excluded. The results in Table 3 clearly demonstrate that the presence of the segmentation branch in the SegDT architecture (40.2% SR, 38.3% SPL) helps improve performance on validation scenes compared to the baseline Decision Transformer (33.4% SR, 29.7% SPL), which does not account for segmentation.

4) ROLE OF THE SEGMENTATION LOSS FUNCTION DURING ONLINE FINE-TUNING

We investigate the impact of the segmentation loss function during online fine-tuning of SegDT. We conduct two

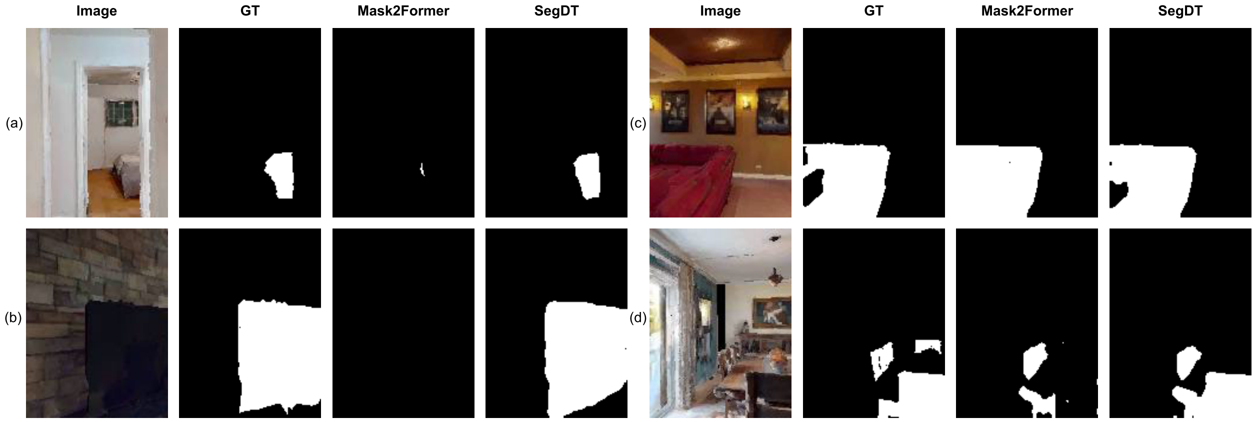


FIGURE 9. Qualitative comparison of segmentation results between SegDT and the baseline Mask2Former model across diverse object categories and viewpoints. SegDT demonstrates superior performance in maintaining consistent object segmentation by leveraging temporal information, particularly evident in cases of partial occlusion and varying object scales.

TABLE 4. Analysis of the Role of the Segmentation Loss Function During Online Fine-Tuning. We use single-frame Mask2Former segmentation for experiments where the segmentation loss function is not included in the training objective during online fine-tuning. The results demonstrate the crucial role of the segmentation loss function during online fine-tuning in improving navigation performance with predicted segmentation.

Method	Navigation semantics	Freeze Image Encoder & Pixel Decoder	Segmentation loss	SR	SPL	SoftSPL
SegDT	Mask2Former	no	no	33.5	30.0	45.3
SegDT	Mask2Former	yes	no	36.8	33.7	47.8
SegDT	**SegDT**	**no**	**yes**	**40.2**	**38.3**	**51.5**

TABLE 5. Analysis of behavioral cloning performance using different sources of ground truth trajectories. Comparison between classical shortest path following and RNN-based goal reaching demonstrates the importance of expert demonstration quality in offline pretraining. Metrics indicate significant advantages of learning from policy-generated trajectories over geometric planning approaches.

GT trajectories source	SR	SPL	SoftSPL
Shortest Path Follower	8.0	6.7	27.3
RNN-based GoalReacher skill [27]	**18.0**	**16.3**	**33.9**

experiments: in the first, we fully unfreeze the model, allowing maximum optimization of its parameters for navigation. In the second, we keep the parts of SegDT responsible for encoding the RGB image frozen—specifically, the Image Encoder and Pixel Decoder. Table 4 shows that freezing the Image Encoder and Pixel Decoder positively affects navigation performance on validation scenes in the absence of a segmentation loss and when using predicted segmentation (36.8% SR vs. 33.5% SR, 33.7% SPL vs. 30.0% SPL). However, incorporating the segmentation loss function during online fine-tuning significantly improves navigation performance when using segmentation predicted by SegDT (40.2% SR vs. 36.8% SR, 38.3% SPL vs. 33.7% SPL).

5) OFFLINE TRAINING ANALYSIS

Table 5 examines the impact of different demonstration sources for behavioral cloning. Using shortest path follower trajectories results in poor performance (8.0% SR, 6.7% SPL). Training on RNN-based goal reacher demonstrations significantly improves results (18.0% SR, 16.3% SPL), highlighting the importance of high-quality expert demonstrations.

TABLE 6. Integration analysis of SegDT within complete ObjectNav systems. Results demonstrate the effectiveness of combining PIRLNav exploration with SegDT goal reaching compared to single-model approaches. Performance gains in both success rate and path efficiency highlight the benefits of specialized skill decomposition.

Exploration	GoalReacher	SR	SPL	SoftSPL
PIRLNav	PIRLNav	61.9	26.0	28.3
PIRLNav	**SegDT**	**63.6**	**30.3**	**32.6**

6) INTEGRATION WITH COMPLETE OBJECTNAV SYSTEMS

Finally, Table 6 shows how SegDT can enhance existing ObjectNav systems. Using PIRLNav for both exploration and goal reaching achieves 61.9% SR with 26.0% SPL. Replacing PIRLNav's goal reaching with SegDT improves performance to 63.6% SR and 30.3% SPL, demonstrating the effectiveness of our specialized goal-reaching approach.

These results collectively demonstrate that SegDT's unified approach to segmentation and navigation, combined with its efficient training strategy, enables robust goal-reaching performance while maintaining computational tractability.

F. VISUALIZATION

Figure 9 demonstrates the qualitative effect of improving segmentation using SegDT for different categories of target objects. The main effect is expressed in filling segmentation gaps if the target object was present in previous frames. The

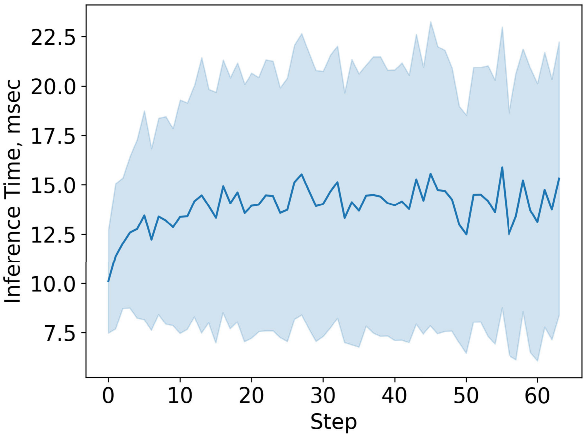


FIGURE 10. Dependence of per-step inference time on the step index, measured on an Nvidia RTX 3060 GPU. The average inference time is shown as a blue line, and the standard deviation is indicated by the light blue shaded area for each step.

aggregation of information from several frames improves the quality of the instantaneous predicted mask contours.

G. INFERENCE TIME

To keep our architecture lightweight and suitable for deployment on mobile onboard platforms, we use a lightweight ResNet50 backbone for Mask2Former, as well as a low image resolution (160 × 120 pixels). Additionally, the computed observation features from one step can be reused for inference at subsequent steps. During inference, we use a history of up to 64 steps. Figure 10 shows the average inference time per step in the sequence, along with its standard deviation, measured on an RTX 3060. Thanks to the efficient reuse of observation features, the per-step inference time does not exceed 25 ms, which corresponds to an inference speed of 40 FPS. Overall, during inference, SegDT uses 3.8 GB of video RAM, which, combined with its inference speed, makes it suitable for deployment on real-time robotic platforms.

VI. CONCLUSION

In this work, we introduced SegDT, a novel transformer-based architecture that unifies semantic segmentation and navigation for embodied agents. Our results demonstrate that jointly optimizing these traditionally separate tasks leads to improved performance in both domains - achieving a 40.2% success rate and 38.3% SPL with predicted segmentation on the goal reacher task, approaching the performance of systems using ground truth semantic information (47.3% SR, 44.7% SPL).

The key innovation of our approach lies in its two-phase training strategy that addresses fundamental limitations of transformer-based navigation. By combining offline pre-training with online policy fine-tuning through knowledge transfer from an RNN-based policy, we achieve robust performance while maintaining computational tractability. Our empirical results show that this approach not only improves segmentation quality (53.7% mIoU vs 51.8% baseline) but also enables more efficient navigation compared to existing methods that treat segmentation as merely an auxiliary task.

While our method demonstrates significant advantages, there remain opportunities for improvement. The computational overhead of transformer-based architectures presents challenges for real-time deployment, and the reliance on pre-trained Mask2Former components could potentially limit generalization to novel environments. Future work could explore more efficient architectures and end-to-end training approaches that eliminate the need for pre-trained components. Another direction for future work is the adaptation of the SegDT approach to the open-vocabulary object goal navigation setting, in particular by using open-vocabulary segmentation models such as OpenSeeD [30] or OV-SAM [61] as backbone segmentation models.

Additionally, investigating methods for selective frame processing during training could further improve both computational efficiency and learning effectiveness. Despite these limitations, our results suggest that unified approaches to perception and action, as demonstrated by SegDT, represent a promising direction for developing more capable embodied agents.

REFERENCES

[1] M. Savva, A. Kadian, O. Maksymets, Y. Zhao, E. Wijmans, B. Jain, J. Straub, J. Liu, V. Koltun, J. Malik, D. Parikh, and D. Batra, "Habitat: A platform for embodied AI research," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9338–9346.

[2] E. Kolve, R. Mottaghi, W. Han, E. VanderBilt, L. Weihs, A. Herrasti, M. Deitke, K. Ehsani, D. Gordon, Y. Zhu, A. Kembhavi, A. Gupta, and A. Farhadi, "AI2-THOR: An interactive 3D environment for visual AI," 2017, *arXiv:1712.05474*.

[3] B. Miao, M. Bennamoun, Y. Gao, and A. Mian, "Region aware video object segmentation with deep motion modeling," *IEEE Trans. Image Process.*, vol. 33, pp. 2639–2651, 2024.

[4] H. Kim, S. Lee, H. Kang, and S. Im, "Offline-to-online knowledge distillation for video instance segmentation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2024, pp. 158–167.

[5] D. S. Chaplot, D. Gandhi, A. Gupta, and R. Salakhutdinov, "Object goal navigation using goal-oriented semantic exploration," in *Proc. NeurIPS*, 2020, pp. 4247–4258.

[6] S. Chen, T. Chabal, I. Laptev, and C. Schmid, "Object goal navigation with recursive implicit maps," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2023, pp. 7089–7096.

[7] J. Shang and M. S. Ryoo, "Active vision reinforcement learning under limited visual observability," in *Proc. NeurIPS*, 2023, pp. 10316–10338.

[8] J. Su, R. Yin, S. Zhang, and J. Luo, "Motion-state alignment for video semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2023, pp. 3571–3580.

[9] J. Šarić, S. Vražić, and S. Šegvić, "Dense semantic forecasting in video by joint regression of features and feature motion," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 9, pp. 6443–6455, Sep. 2023.

[10] C. Graber, C. Jazra, W. Luo, L. Gui, and A. Schwing, "Joint forecasting of panoptic segmentations with difference attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2022, pp. 2558–2567.

[11] J. Zhang, L. Dai, F. Meng, Q. Fan, X. Chen, K. Xu, and H. Wang, "3D-aware object goal navigation via simultaneous exploration and identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 6672–6682.

[12] W. Yingcai and L. Fang, "Joint 2D and 3D semantic segmentation with consistent instance semantic," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. 107, no. 8, pp. 1309–1318, Aug. 2024.

[13] G. Scarpellini, S. Rosa, P. Morerio, L. Natale, and A. D. Bue, "Look around and learn: Self-training object detection by exploration," in *Proc. ECCV*, 2024, pp. 72–88.

[14] P. Tian, M. Yao, X. Xiao, B. Zheng, T. Cao, Y. Xi, H. Liu, and H. Cui, "3-D semantic terrain reconstruction of monocular close-up images of Martian terrains," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–16, 2024.

[15] S. Zhu, R. Qin, G. Wang, J. Liu, and H. Wang, "SemGauss-SLAM: Dense semantic Gaussian splatting SLAM," 2024, *arXiv:2403.07494*.

[16] X. Lei, M. Wang, W. Zhou, and H. Li, "GaussNav: Gaussian splatting for visual navigation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 5, pp. 4108–4121, May 2025.

[17] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.

[18] T. B. Brown et al., "Language models are few-shot learners," in *Proc. NeurIPS*, 2020, pp. 1877–1901.

[19] H. Zhang, H. You, P. Dufter, B. Zhang, C. Chen, H.-Y. Chen, T.-J. Fu, W. Y. Wang, S.-F. Chang, Z. Gan, and Y. Yang, "Ferret-v2: An improved baseline for referring and grounding with large language models," 2024, *arXiv:2404.07973*.

[20] C. Lu, R. Shi, Y. Liu, K. Hu, S. S. Du, and H. Xu, "Rethinking transformers in solving POMDPs," in *Proc. ICML*, 2024, pp. 33089–33112.

[21] T. Ni, M. Ma, B. Eysenbach, and P. Bacon, "When do transformers shine in RL? Decoupling memory from credit assignment," in *Proc. NeurIPS*, 2023, pp. 50429–50452.

[22] 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," 2020, *arXiv:2006.13171*.

[23] S. K. Ramakrishnan, D. S. Chaplot, Z. Al-Halah, J. Malik, and K. Grauman, "PONI: Potential functions for ObjectGoal navigation with interaction-free learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 18868–18878.

[24] N. Yokoyama, S. Ha, D. Batra, J. Wang, and B. Bucher, "VLFM: Vision-language frontier maps for zero-shot semantic navigation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2024, pp. 42–48.

[25] N. Yokoyama, R. Ramrakhya, A. Das, D. Batra, and S. Ha, "HM3D-OVON: A dataset and benchmark for open-vocabulary object goal navigation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2024, pp. 5543–5550.

[26] O. Maksymets, V. Cartillier, A. Gokaslan, E. Wijmans, W. Galuba, S. Lee, and D. Batra, "THDA: Treasure hunt data augmentation for semantic navigation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 15354–15363.

[27] A. Staroverov, K. Muravyev, K. Yakovlev, and A. I. Panov, "Skill fusion in hybrid robotic framework for visual object goal navigation," *Robotics*, vol. 12, no. 4, p. 104, Jul. 2023.

[28] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 1280–1289.

[29] J. Jain, J. Li, M. Chiu, A. Hassani, N. Orlov, and H. Shi, "OneFormer: One transformer to rule universal image segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 2989–2998.

[30] H. Zhang, F. Li, X. Zou, S. Liu, C. Li, J. Yang, and L. Zhang, "A simple framework for open-vocabulary segmentation and detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 1020–1031.

[31] P. Wang, Z. Cai, H. Yang, A. Swaminathan, R. Manmatha, and S. Soatto, "Mixed-query transformer: A unified image segmentation architecture," 2024, *arXiv:2404.04469*.

[32] D. Morilla-Cabello, L. Mur-Labadia, R. Martinez-Cantin, and E. Montijano, "Robust fusion for Bayesian semantic mapping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2023, pp. 76–81.

[33] Y. Tao, X. Liu, I. Spasojevic, S. Agarwal, and V. Kumar, "3D active metric-semantic SLAM," *IEEE Robot. Autom. Lett.*, vol. 9, no. 3, pp. 2989–2996, Mar. 2024.

[34] T. Zemskova, A. Staroverov, K. Muravyev, D. A. Yudin, and A. I. Panov, "Interactive semantic map representation for skill-based visual object navigation," *IEEE Access*, vol. 12, pp. 44628–44639, 2024.

[35] B. Yu, H. Kasaei, and M. Cao, "Frontier semantic exploration for visual target navigation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2023, pp. 4099–4105.

[36] T. Zhang, X. Tian, Y. Zhou, S. Ji, X. Wang, X. Tao, Y. Zhang, P. Wan, Z. Wang, and Y. Wu, "DVIS++: Improved decoupled framework for universal video segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 7, pp. 5918–5929, Jul. 2025.

[37] I. Shin, D. Kim, Q. Yu, J. Xie, H.-S. Kim, B. Green, I. S. Kweon, K.-J. Yoon, and L.-C. Chen, "Video-kMaX: A simple unified approach for online and near-online video panoptic segmentation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2024, pp. 228–238.

[38] G. Zhou, Y. Hong, and Q. Wu, "NavGPT: Explicit reasoning in vision-and-language navigation with large language models," in *Proc. AAAI Conf. Artif. Intell.*, 2024, vol. 38, no. 7, pp. 7641–7649.

[39] K. Kotar and R. Mottaghi, "Interactron: Embodied adaptive object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 14860–14869.

[40] L. Fan, M. Liang, Y. Li, G. Hua, and Y. Wu, "Evidential active recognition: Intelligent and prudent open-world embodied perception," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 16351–16361.

[41] W. Ding, N. Majcherczyk, M. Deshpande, X. Qi, D. Zhao, R. Madhivanan, and A. Sen, "Learning to view: Decision transformers for active object detection," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2023, pp. 7140–7146.

[42] J. Yang, Z. Ren, M. Xu, X. Chen, D. Crandall, D. Parikh, and D. Batra, "Embodied amodal recognition: Learning to move to perceive objects," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 2040–2050.

[43] H. Zhao, M. Yao, Y. Zhao, Y. Jiang, H. Zhang, X. Xiao, and K. Gao, "M2CS: A multimodal and campus-scapes dataset for dynamic SLAM and moving object perception," *J. Field Robot.*, vol. 42, no. 3, pp. 787–805, May 2025.

[44] T. Zemskova, M. Kichik, D. Yudin, A. Staroverov, and A. Panov, "SegmATRon: Embodied adaptive semantic segmentation for indoor environment," *Neurocomputing*, vol. 638, Jul. 2025, Art. no. 130169.

[45] K. Yadav, R. Ramrakhya, S. K. Ramakrishnan, T. Gervet, J. Turner, A. Gokaslan, N. Maestre, A. X. Chang, D. Batra, M. Savva, A. W. Clegg, and D. S. Chaplot, "Habitat-matterport 3D semantics dataset," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Vancouver, BC, Canada, Jun. 2023, pp. 4927–4936.

[46] Y. Xiong, X. Xiao, M. Yao, H. Liu, H. Yang, and Y. Fu, "Mars-Former: Martian rock semantic segmentation with transformer," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–12, 2023.

[47] H. Liu, M. Yao, X. Xiao, and Y. Xiong, "RockFormer: A U-shaped transformer network for Martian rock segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–16, 2023.

[48] Y. Xiong, X. Xiao, M. Yao, H. Cui, and Y. Fu, "Light4Mars: A lightweight transformer model for semantic segmentation on unstructured environment like Mars," *ISPRS J. Photogramm. Remote Sens.*, vol. 214, pp. 167–178, Aug. 2024.

[49] H. Liu, M. Yao, X. Xiao, B. Zheng, and H. Cui, "MarsScapes and UDAFormer: A panorama dataset and a transformer-based unsupervised domain adaptation framework for Martian terrain segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–17, 2024.

[50] Y. Hong, Y. Zhou, R. Zhang, F. Dernoncourt, T. Bui, S. Gould, and H. Tan, "Learning navigational visual representations with semantic map supervision," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 3032–3044.

[51] L. Chen, K. Lü, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch, "Decision transformer: Reinforcement learning via sequence modeling," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 15084–15097.

[52] X. B. Peng, A. Kumar, G. Zhang, and S. Levine, "Advantage-weighted regression: Simple and scalable off-policy reinforcement learning," 2019, *arXiv:1910.00177*.

[53] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niebner, M. Savva, S. Song, A. Zeng, and Y. Zhang, "Matterport3D: Learning from RGB-D data in indoor environments," in *Proc. Int. Conf. 3D Vis. (3DV)*, Oct. 2017, pp. 667–676.

[54] K. Yadav, J. Krantz, R. Ramrakhya, S. K. Ramakrishnan, J. Yang, A. Wang, J. Turner, A. Gokaslan, V.-P. Berges, R. Mootaghi, O. Maksymets, A. X. Chang, M. Savva, A. Clegg, D. S. Chaplot, and D. Batra. (2023). *Habitat Challenge*. [Online]. Available: https://aihabitat.org/challenge/2023/

[55] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2017, *arXiv:1711.05101*.

[56] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, "High-dimensional continuous control using generalized advantage estimation," 2015, *arXiv:1506.02438*.

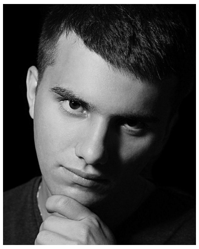
[57] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[58] A. Kumar, S. Gupta, D. F. Fouhey, S. Levine, and J. Malik, "Visual memory for robust path following," in *Proc. NeurIPS*, vol. 31, 2018, pp. 765–774.

[59] E. Wijmans, A. Kadian, A. S. 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 *Proc. ICLR*, 2019, pp. 6–59.

[60] R. Ramrakhya, D. Batra, E. Wijmans, and A. Das, "PIRLNav: Pretraining with imitation and RL finetuning for OBJECTNAV," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 17896–17906.

[61] H. Yuan, X. Li, C. Zhou, Y. Li, K. Chen, and C. C. Loy, "Open-vocabulary SAM: Segment and recognize twenty-thousand classes interactively," in *Proc. Eur. Conf. Comput. Vis.*, 2024, pp. 419–437.



ALEKSEI STAROVEROV received the M.S. degree from Bauman Moscow State Technical University, Moscow, Russia, in 2019. He is currently pursuing the Ph.D. degree in computer science with Moscow Institute of Physics and Technology, Moscow. His research thesis involves the methods and algorithms for the automatic determination of subgoals in a reinforcement learning problem for robotic systems. Since 2022, he has been a Researcher with the Artificial Intelligence Research Institute, Moscow. His research interests include reinforcement learning, deep learning, and robotic systems.

TATIANA ZEMSKOVA received the M.S. degree in applied mathematics and computer science from Moscow Institute of Physics and Technology, Moscow, Russia, in 2023 and the M.S. degree in engineering from the École Polytechnique, Palaiseau, France, in 2023. She is currently pursuing the Ph.D. degree in computer science with Moscow Institute of Physics and Technology. Since 2024, she has been a Junior Research Scientist with the Artificial Intelligence Research Institute, Moscow. Her research interests include computer vision, embodied AI, and robotic systems.

DMITRY A. YUDIN received the Engineering Diploma degree in automation of technological processes and production and the Ph.D. degree in computer science from Belgorod State Technological University (BSTU), named after V. G. Shukhov, Belgorod, Russia, in 2010 and 2014, respectively. From 2009 to 2019, he was a Researcher and an Assistant Professor with the Technical Cybernetics Department, BSTU n.a. V. G. Shukhov. Since 2019, he has been the Head of the Intelligent Transport Laboratory, Moscow Institute of Physics and Technology, Moscow, Russia. Since 2021, he has been a Senior Researcher with the Artificial Intelligence Research Institute (AIRI), Moscow. He is the author of more than 100 articles. His research interests include computer vision, deep learning, and robotics.

ALEKSANDR I. PANOV (Member, IEEE) received the Ph.D. degree in theoretical computer science from the Institute for Systems Analysis, Moscow, Russia, in 2015, and the Dr.Sc. degree in artificial intelligence from Moscow Institute of Physics and Technology, Moscow, in 2024. Since 2019, he has been the Head of the Center for Cognitive Modeling, Moscow Institute of Physics and Technology. In 2021, he joined the Research Group on Cognitive AI Systems, Artificial Intelligence Research Institute, Moscow. He has authored three books and more than 200 research papers. His academic interests include behavior planning, reinforcement learning, embodied AI, and cognitive robotics.