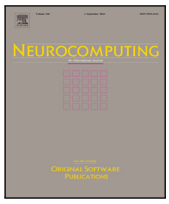
Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom



Data availability

Links listed below are the data deposition URLs from Data availability section. Please verify the links are valid. This page will not appear in the article PDF file or print. They are displayed in the proof pdf for review purpose only.

https://github.com/wingrune/SegmATRon Dataset Link: https://github.com/wingrune/SegmATRon

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

SegmATRon: Embodied adaptive semantic segmentation for indoor environment

Tatiana Zemskova [a,b,*], Margarita Kichik [a], Dmitry Yudin [a,b], Aleksei Staroverov [b], Aleksandr Panov [a,b,c]

[a] Moscow Institute of Physics and Technology, 9 Institutsky per., Dolgoprudny, 141701, Russia
[b] AIRI, 32 Kutuzovsky Ave., Moscow, 121170, Russia
[c] Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, 9 60-Letiya Oktyabrya Ave., Moscow, 117312, Russia

ARTICLE INFO

Communicated by A. Loddo

Dataset link: https://github.com/wingrune/SegmATRon

Keywords:
Semantic segmentation
Embodied vision
Active vision
Transformer
Indoor environment
Simulation

ABSTRACT

The state-of-the-art methods for computer vision are often trained with large amounts of data collected from static cameras. In contrast, an embodied intelligent agent can interact with a continuous environment to improve the perception quality. Previous methods for embodied computer vision have not considered the task of semantic segmentation. This paper first introduces an adaptive transformer model for embodied image semantic segmentation named SegmATRon. Its distinctive feature is the adaptation of model weights during inference on several images using a hybrid multicomponent loss function. We studied this model on datasets collected in the photorealistic Habitat and the synthetic AI2-THOR simulators. We showed that obtaining additional images using the agent's actions in an indoor environment can improve the quality of semantic segmentation.

1. Introduction

Embodied Artificial Intelligence involves studying agents that can solve intellectual tasks while interacting with the environment autonomously [1,2].

This is especially important for modern robots, which must perform reliable scene recognition using onboard sensors (usually cameras) while simultaneously performing navigation or object manipulation tasks [3–5].

Recently, embodied methods in object detection [6–9] have appeared, which demonstrate that the information fusion from an image sequence during indoor navigation positively affects the quality of detection. However, the existing embodied approaches do not consider semantic segmentation, another important perception task for intelligent agents [10].

Training semantic segmentation neural networks requires laborious work of class annotation for every image pixel. An embodied agent is meant to navigate through different indoor environments; therefore, its deployment would be delayed by the necessity of collecting additional data to fine-tune its segmentation module. A promising solution for collecting and pre-labeling additional data may be embodied in semantic segmentation methods. They allow the choice of agent movement policy to maximize segmentation accuracy similar to the interactive object detection approach [7].

Annotations for collected images also can be obtained by pseudo-labeling with accumulated semantic 3D maps [11] or requested from human experts [12–14]. At the same time, an adaptive learning technique [15] can be used in robot visual navigation tasks to improve domain adaptation by unfreezing the model during inference. This technique allows the agent to perform its initial task directly without additional exploration, aiming to collect data for fine-tuning. Inspired by work [7], we propose and investigate an adaptive learning method with different action policies for the improvement of semantic segmentation in the Habitat [16] and AI2-THOR [17] indoor environments. These environments are among the most popular for researching the problems of interactive perception and navigation of embodied agents.

We propose the SegmATRon architecture, which adapts state-of-the-art semantic segmentation models, Mask2Former [18] and MaskDINO [19], trained on static frames, to the interactive perception setting. The SegmATRon method fuses information from multiple frames using two mechanisms.

First, SegmATRon employs a transformer-based Fusion Module that aggregates predictions and image features obtained from a sequence

* Corresponding author at: Moscow Institute of Physics and Technology, 9 Institutsky per., Dolgoprudny, 141701, Russia.
E-mail addresses: zemskova.ts@phystech.su (T. Zemskova), margarita.kichik@gmail.com (M. Kichik), yudin.da@mipt.ru (D. Yudin), staroverov@airi.net (A. Staroverov), panov.ai@mipt.ru (A. Panov).

https://doi.org/10.1016/j.neucom.2025.130169
Received 21 September 2024; Received in revised form 19 March 2025; Accepted 29 March 2025

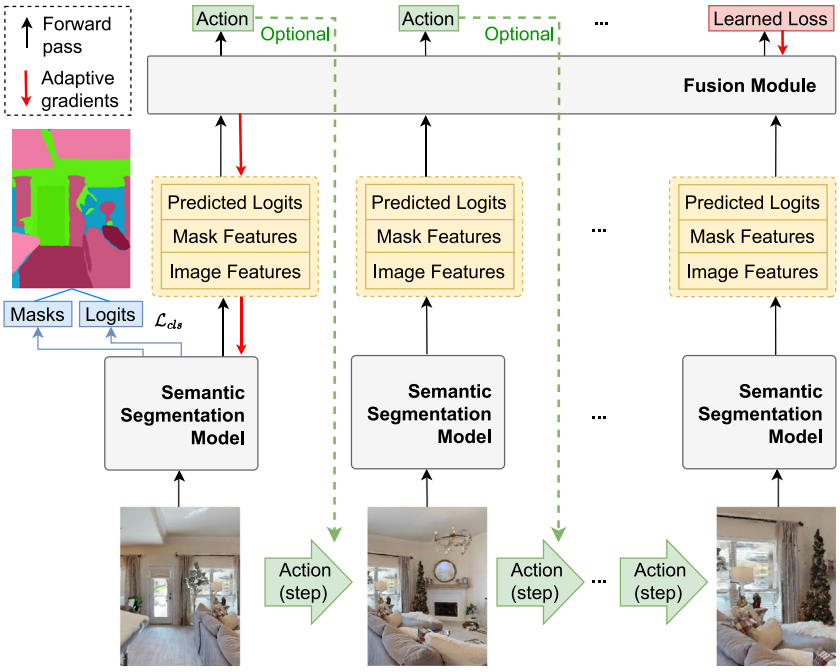


Fig. 1. Simplified inference scheme of the proposed SegmATRon approach. Adapting the Semantic Segmentation Model weights during inference on several images is made via learned loss predicted by the Fusion Module to improve the segmentation quality of the first frame. The Transformer-based Fusion Module inputs are predicted semantic logits, mask features, and image features from the Semantic Segmentation Model. The Fusion module outputs predicted learned loss and, optionally, action. The action can be used to choose the next frame. The Fusion module infers the learned loss when the necessary number of frames is collected.

of images. Second, SegmATRon utilizes a multi-component hybrid loss function that includes an adaptive component predicted by the model itself. This adaptive loss function is used both during training and inference to adjust the weights of the segmentation model, improving segmentation quality on the first image in the sequence.

Finally, we conduct experiments on learning an action policy for collecting new frames to the sequence, demonstrating the ability to achieve more stable segmentation improvements compared to a random policy.

To summarize, our contributions are the following:

- We have proposed a new architecture for an embodied adaptive semantic segmentation neural network called *SegmATRon* (see Fig. 1). In contrast to the state-of-the-art methods for semantic segmentation, the SegmATRon actively collects additional images from the environment to improve the semantic segmentation quality.
- We have developed a transformer *Fusion module* that takes image and mask features, predicted semantic logits, and masks as inputs and generates output actions that an intelligent agent can perform in the environment to obtain new images.
- We have proposed to use the multicomponent hybrid loss function involving adaptive learned loss, which value is predicted by the SegmATRon. This loss value is then used in the inference to adapt the basic semantic segmentation model. It leads to an increase in the segmentation quality of the first image of the sequence.
- To study the quality metrics of embodied semantic segmentation, we have created two novel datasets based on the Habitat and AI2-THOR simulators. These datasets contain not only images and masks for semantic segmentation but also a tree of possible actions that an agent can perform from some point in indoor scenes. Thus, we demonstrate the possibility of using our approach in a multi-embodied mode.

The code of the proposed approach and datasets are publicly available at https://github.com/wingrune/SegmATRon.

2. Related works

Image Semantic Segmentation. To address the semantic segmentation task, methods based on CNNs and more recent transformer-based approaches have been developed.

The newest but CNN-based foundation model InternImage [20] and large HRNet-based [21] methods with attention mechanisms like HRNet+OCR [22] and HRNetV2-OCR+PSA [23] belong to the first category.

Transformer-based OneFormer [24] belongs to the second category. It outperforms other state-of-the-art methods, such as Mask2Former [18], k-means Mask Transformer [25], and Panoptic-Deeplab [26] in solving tasks of semantic, instance, and panoptic segmentation. Notably, these achievements are attained without needing to train separately for each task.

In transformer-based approaches such as OneFormer [24] and Mask2Former [18], learnable query vectors are fed into the transformer mask decoder. The authors of CLUSTSEG [27] propose an improvement to this approach by introducing task-specific query initialization, as well as iterative clustering and centroid updates during mask decoding. Another approach to decoding masks from features extracted by the backbone is the creation of non-learnable class prototypes, as presented in [28]. In this case, a nonparametric nearest prototype search is used to generate dense segmentation mask predictions.

Recently, the foundation model Segment Anything (SAM) [29] has gained popularity for image segmentation tasks. However, this model does not suit semantic segmentation because SAM predicts the segmentation masks in a class-agnostic manner.

One of the important traits of embodied computer vision methods is the need for adaptation to different domains. One approach to adapt a method to a new domain, where semantic segmentation annotations are unavailable, is Unsupervised Domain Adaptation. Existing works [30] propose unsupervised learning on an unannotated target domain dataset along with modules that extract discriminative features between categories across domains. In the work [31], the issue of overfitting on the source domain is addressed, and a method for

extracting hybrid domain features and a learning approach to improve generalization to a new domain are proposed. The authors of [32] suggest an approach for reweighting labeled examples from the source domain based on the global distribution of source and target domains. In the SegmATRon method, the domain adaptation is done via an adaptive loss function. Experiments show that adaptation during inference allows effective segmentation on a new domain without additional fine-tuning making our method more suitable for intelligent agents.

Information Fusion for Image Segmentation. To improve segmentation quality, additional information beyond a single RGB image can be used. For example, the authors of [33] propose a learning method where images in the training dataset containing the same semantic class can be used to create a pixel-wise contrastive learning signal, improving the quality of pixel embeddings belonging to the same semantic class. Another type of such information is the modalities from other sensors that may be installed on board an embodied intelligent agent. Common types of these sensors include depth and thermal sensors.

For instance, the authors of MFFENet [34] use the combination of multi-scale features extracted from RGB images and thermal maps to improve urban road scene parsing. FRNet [35] proposes a method to enhance the representation quality of features by multi-level fusing of RGB and depth images. In the MTANet [36] paper, in addition to the multi-level fusion of RGB and thermal features, simultaneous training on several types of segmentation tasks is used to improve segmentation quality. The LSNet [37] work presents a lightweight architecture to enhance the speed of multimodal RGB-T salient object detection, along with a boundary computation algorithm and the use of transfer learning to improve feature generation quality. The method in [38] uses a wavelet-based MLP for feature extraction in RGBT images as well as knowledge distillation techniques to improve training quality.

The use of additional modalities makes the method sensitive to changes in data quality when transitioning from simulated data to real-world domains. The authors of [39] propose gradual feature fusion and a module to improve depth map quality using information contained in RGB images. In the method [40], multi-level features from images and depth maps are additionally used to improve the quality of extracted features. These methods use RGB image features to improve features extracted from depth maps. Meanwhile, the BCINet method [41] presents modules for mutual enhancement of features extracted from RGB and depth data.

However, depending on the sensor, depth can be represented either as a map or as a sparse point cloud. One of the features of SegmATRon is its ability to adapt to different types of environments, which is why in our work, we consider only one sensor modality—RGB images from a camera. This makes our method related to video segmentation approaches.

Video Segmentation. An embodied agent receives information about an environment through a frame sequence. Classical Computer Vision methods, which do not consider camera movement, solve the task of frame sequence segmentation in the scope of Video Segmentation. Recently, densely annotated benchmarks such as CityScapes-VPS [42], VIPSeg [43], and VIPOSeg [44] have appeared, which led to the emergence of video instance segmentation methods.

TarVIS [45] is flexible for solving segmentation and detection tasks, MaskFreeVIS [46] does not use masks for training, DVIS [47] implements the decoupling strategy for video instance segmentation, Video-kMaX [48] bridges the gap between online and offline video segmentation methods. These and other methods are capable of predicting a category for every pixel of video frames.

Existing video segmentation methods propose various approaches for utilizing the mask from the first frame to segment subsequent frames. For example, in [49], a method for self-supervised video segmentation is introduced, which learns mask embeddings from unlabeled videos using pseudo-labels and also proposes a learning method for short-term and long-term correspondences between visual features of different frames. SAM2 [50] is a foundation model for video segmentation that employs a lightweight image encoder and memory attention to extract information about the target object from previous frames. A drawback of such approaches is the requirement for a high-quality prompt mask in the initial frame, which is then propagated, as well as their class-agnostic nature. This setup is not suitable for deploying a segmentation module onboard a mobile robot to extract information about the semantics of the environment.

A distinguishing feature of our method compared to methods for video segmentation is the adaptive loss function facilitating the model adaptation across different indoor environments without fine-tuning. Furthermore, the mentioned methods require a sequence of frames to be provided, whereas our approach uses only 5 frames acquired from distinct domains. Finally, we show that SegmATRon can predict actions to collect additional frames, further improving segmentation quality, which is not possible for methods that treat the video segmentation task with a static camera (See Section 6).

Embodied Computer Vision. Several environments simulating living spaces have been developed for embodied agents, including Habitat [51] and AI2-THOR [17], enabling navigation within the environment and object interactions. A wide range of embodied computer vision methods is present in the field.

The recent work [9] proposes to learn a policy for navigation that maximizes the confidence score of a frozen object detector. [6,52] learn to maximize segmentation quality by selecting the next best view based on image features derived from neural network models, whereas [53] demonstrates that a voting system based on four criteria derived from the initial viewpoint can improve object recognition. [8, 54], and [55] exploit different policies for push actions to increase the quality of instance segmentation for an embodied agent with a gripper. Another important perception task for intelligent agents is scene semantic segmentation. At the moment, for this task, there are no active methods that allow the perception neural network to control the agent movement to improve scene recognition quality.

Active exploration is crucial in developing embodied agents capable of acting in complex or unfamiliar environments. Examples of such agents include Ask4Help [56], which uses human expert hints, and Move to See Better [57], which uses multiple frames for fine-tuning during testing. SEAL [58] uses a sequence of images and depth maps to aggregate multi-view semantic information into a 3D map using self-supervised label propagation. Unlike the works [57] and [58], our method does not require depth maps as the information fusion is done in latent space.

Another instance of an active embodied agent is the Interactron [7], which involves continuous fine-tuning of the detector model during inference. A supervisor is incorporated into the model to adjust the detector's parameters and determine the action policy. The agent navigates through the environment, executing actions from the predetermined set of actions. A notable feature of the Interactron is its adaptive loss function.

Our work applies a similar approach to address the semantic segmentation task. We introduce a new set of actions and demonstrate that executing just a single additional action is sufficient to enhance segmentation quality.

The adaptive learned loss function in our method improves the model quality and its ability to generalize to unseen environments. Another strategy for effectively retraining computer vision models in the environment is to collect data based on feedback from the computer vision model. Our method presents the advantage of facilitating adaptation to new domains without necessitating further retraining, along with subsequent inference to improve semantic segmentation quality.

3. Method

We formulate the embodied semantic segmentation problem as follows. An agent spawns randomly within an unfamiliar environment and receives an RGB observation F_0. Then, the agent collects additional RGB observations $\{F_1, \dots, F_N\}, N \in (1, 2, 3, 4)$ using the policy π. The collected frame sequence $\{F_0, \dots, F_N\}$ is fed as input to the agent, which predicts the multichannel semantic segmentation mask for the initial frame M_0 by aggregating information from the frame sequence using an adaptive loss function. In our experiments, we use the following action space: turn left, turn right by angle 30°, look up, look down by tilting the agent head by angle 30°, and move backward by 0.25 m.

The SegmATRon architecture consists of two modules: a semantic segmentation model and the Fusion module. In our experiments, we consider two segmentation models, Mask2Former [18] and MaskDINO [19]. The role of the Fusion module is to aggregate information from multiple frames and use it to control segmentation in two ways. First, the Fusion module predicts the value of the learned loss function, which is used to adapt the weights of the segmentation model during inference. Secondly, the Fusion model is used to predict an action to collect the next frame.

Adaptive Learning. The key idea of adaptive semantic segmentation is to train the Fusion module to predict an estimate of the segmentation loss function for the first frame of a given frame sequence. Then, during inference, this loss function can be used to change the parameters of the segmentation model for a specific set of observations, improving the segmentation quality.

The adaptation of segmentation model weights during inference on several images is done via a hybrid multicomponent loss function with an adaptive learned part $\mathcal{L}_{adapt}(\phi, \theta, \mathbf{F})$. The loss function is parameterized by Fusion Module parameters ϕ and depends on parameters θ of a segmentation model and a sequence of frames $\mathbf{F}$. The goal during the training process is to minimize the multicomponent loss $\mathcal{L}_{segm}(\theta, \mathbf{F})$ over all ground-truth sequences $\mathbf{R}_{all}$, where the parameters θ are updated by backpropagation through adaptive gradients with a learning rate equal to α:

$$\min_{\theta,\phi} \sum_{\mathbf{F}\in\mathbf{R}_{all}} \mathcal{L}_{segm}(\theta - \alpha\nabla_\theta \mathcal{L}_{adapt}(\phi, \theta, \mathbf{F}), \mathbf{F}). \tag{1}$$

The loss function (1) is optimized iteratively for each mini-batch by first updating the parameters θ using their respective gradients, followed by updating the parameters ϕ using the gradients with respect to ϕ. As the gradients for ϕ depend on the current values of θ, the θ parameters are held constant during the ϕ update, retaining their pre-update values from the current mini-batch.

For each of the considered segmentation models, we use the loss functions proposed by the authors of the respective models. We use the following segmentation loss function for the Mask2Former [18] model:

$$\mathcal{L}_{segm}^{Mask2Former} = \lambda_{cls}\mathcal{L}_{cls} + \lambda_{bce}\mathcal{L}_{bce} + \lambda_{dice}\mathcal{L}_{dice}, \tag{2}$$

where, $\mathcal{L}_{cls}$ is the cross-entropy loss for class prediction. The binary cross-entropy ($\mathcal{L}_{bce}$) and the dice loss ($\mathcal{L}_{dice}$) are controlling mask predictions. We use the set of hyper-parameters proposed in the Mask2Former [18] for segmentation loss $\lambda_{cls} = 2$, $\lambda_{bce} = 5$, and $\lambda_{dice} = 5$. λ_{cls} is set to 0.1 for the no-object prediction.

For the MaskDINO model we use the loss function proposed by Li et al. in the original work [19]:

$$\mathcal{L}_{segm}^{MaskDINO} = \lambda_{cls}\mathcal{L}_{cls} + \lambda_{focal}\mathcal{L}_{focal} + \lambda_{dice}\mathcal{L}_{dice} + \lambda_{L1}\mathcal{L}_{L1} + \lambda_{giou}\mathcal{L}_{giou}, \tag{3}$$

where, $\mathcal{L}_{cls}$ is the cross-entropy loss for class prediction. The focal loss ($\mathcal{L}_{focal}$) and the dice loss ($\mathcal{L}_{dice}$) are controlling mask predictions. The L1-regression loss ($\mathcal{L}_{L1}$) and the GIoU loss ($\mathcal{L}_{giou}$) are used for bounding boxes predictions. We use the same set of hyper-parameters as the authors of MaskDINO: $\lambda_{cls} = 4$, $\lambda_{bce} = 5$, $\lambda_{dice} = 5$, $\lambda_{L1} = 5$, and $\lambda_{giou} = 5$.

During the inference process, when a new sequence of frames $\{F_0, \dots, F_N\}$ arrives, the prediction of a multi-channel semantic segmentation mask occurs in two stages. First, for every frame in the sequence the segmentation model generates image embeddings and masks and logits predictions. Second, the Fusion Module predicts the learned adaptive loss function value $\mathcal{L}_{adapt}(\phi, \theta, \mathbf{F})$. The segmentation model parameters θ for a given observation are then updated using a stochastic gradient descent step with the following formula: $\theta^{adapt} = \theta - \alpha\nabla_\theta \mathcal{L}_{adapt}(\phi, \theta, \mathbf{F})$. The segmentation model with the updated weights θ^{adapt} is used to make the final prediction. After a prediction has been made for the current sequence of frames, the segmentation model weights return to θ until the next observation arrives.

Action Prediction. We adopt an approach similar to the method proposed by the authors of Interactron [7]. During training, the SegmATRon gradually explores possible trajectories by randomly sampling actions and learns to predict the best path from the observed. The path is considered the best if it gives the smallest ground truth weighted segmentation loss.

State-of-the-art segmentation models like Mask2Former [18] require several hundred epochs for training. Therefore, when preserving the best paths for a sequence of 5 frames, a situation may arise where the best path corresponds to a local minimum of the ground-truth loss for certain categories represented in the images. We expect that during training, the model will accumulate a sufficient number of trajectory demonstrations to generalize to validation scenes only for object categories that appear most frequently.

Additionally, we anticipate that the frame selection policy has the greatest impact on foreground categories that exclude the floor, ceilings, and walls. Therefore, we propose the Weighted Best Loss policy. We assign weights to the segmentation loss term responsible for mask class prediction.

To learn to predict the best path from a frame sequence an additional component is added to $\mathcal{L}_{segm}$ in Eq. (1):

$$\min_{\theta,\phi} \sum_{\mathbf{F}\in\mathbf{R}_{all}} \mathcal{L}_{segm}^{weighted}(\theta^{adapt}, \mathbf{F}) + \mathcal{L}_{ce}(p_{pred}, p_{best}), \tag{4}$$

where, $\mathcal{L}_{ce}$ – the cross-entropy loss between the predicted sequence of actions p_{pred} and the best sequence of actions seen so far p_{best}, $\theta^{adapt} = \theta - \alpha\nabla_\theta \mathcal{L}_{adapt}(\phi, \theta, \mathbf{F})$ - the adapted value of the segmentation model parameters θ.

$$\mathcal{L}_{segm}^{weighted} = \lambda_{cls} \sum_{c=1}^{C} w_c \mathcal{L}_{cls}^c + \lambda_{bce}\mathcal{L}_{bce} + \lambda_{dice}\mathcal{L}_{dice}. \tag{5}$$

Here, $c \in \{1, .., C\}$ represents the set of classes in the dataset, and the weight is defined as:

$$w_c = \frac{N_{\text{train}}^c}{N_{\text{train}}}, \tag{6}$$

where N_{train}^c is the number of images containing class c in the training set if $c \notin$ {wall, floor, ceiling}. N_{train} is the number of training images that contain classes $c \notin$ {wall, floor, ceiling}. If $c \in$ {wall, floor, ceiling}, then $w_c = 0$.

We also test a random policy for action selection to collect subsequent frames. In practice, action selection may be driven by other goals, like navigation, rather than improving semantics. Even with random frame sequences, an adaptive loss function utilization enhances the segmentation quality (see Section 6). Therefore we consider action prediction to be optional in our experiments.

Segmentation model. As segmentation models (see Fig. 2), we consider the modification of Mask2Former [18] and MaskDINO [19], which represent state-of-the-art methods for semantic segmentation. The off-the-shelf Mask2Former and MaskDINO use a single frame to make predictions of masks and labels. The off-the-shelf models represent baseline approaches for comparison with our SegmATRon model.

Fusion module. Following the idea of Interactron [7], we choose a Transformer model to combine predictions and image features from

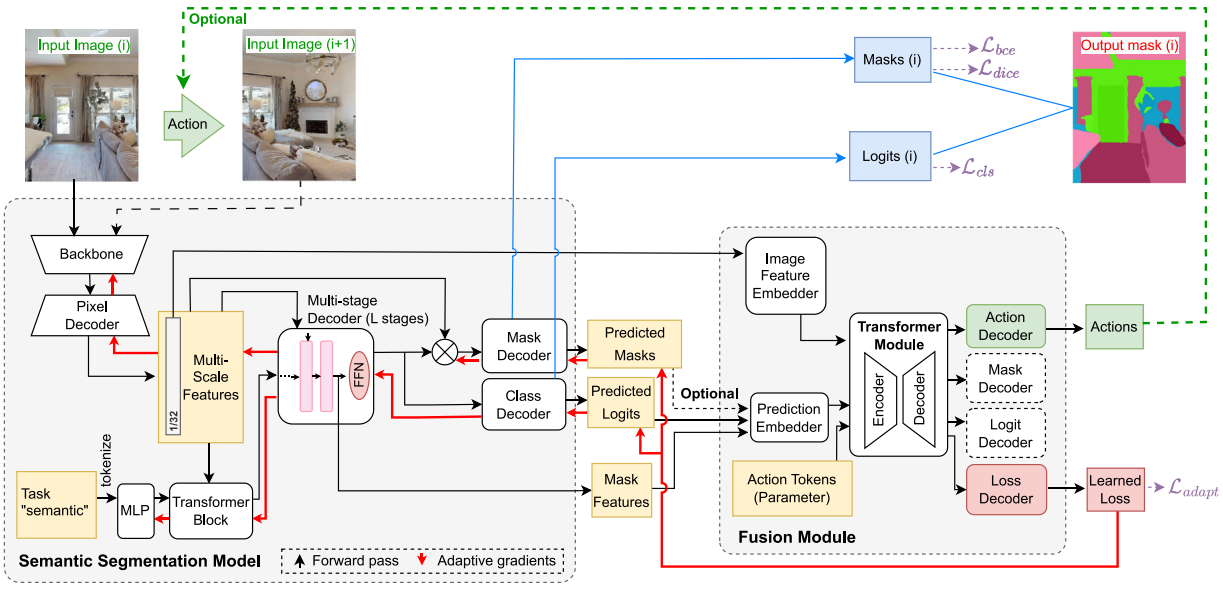


Fig. 2. Detailed scheme of the SegmATRon approach. It includes two main parts: a Semantic Segmentation Model (Mask2Former or MaskDINO) and a Fusion Module. The Semantic Segmentation Model consists of an Image Backbone, Pixel Decoder, Transformer Block, Multi-Stage Decoder, Mask and Class Decoders. The Fusion Module aggregates features and predictions of the Segmentation Model and predicts Actions (optional) and Learned Loss for adaptive inference of SegmATRon. The Fusion Module consists of Image Feature and Prediction Embedders, a Transformer Module, and Decoders for Action, Loss, Logits, and Masks. The output segmentation result is shown in blue color. Also, the diagram shows how various data are involved in calculating the considered loss functions. The red lines show the adaptive gradients flow.

several frames to predict the loss for the adaptive backward pass. As the Transformer Model, we use the GPT [59] designed to handle sequences of variable length.

The Fusion Module (see Fig. 2) takes as an input the 1/32 feature map from the Multi-scale Pixel Decoder of the segmentation model, predicted logits of mask classification, and mask features. Mask features are represented by the input of the last FFN layer of the last stage of the segmentation model Multi-stage Decoder. This input is mapped to the dimension of the Transformer module by corresponding embedders. We change the architecture of the Prediction Embedder in the Fusion Module compared to the Fusion Module provided by the authors of Interactron [7]. Previous works [60] show the advantage of non-linear projection heads for the selection of a subset of features to apply the contrastive loss function in self-supervised tasks. Following this observation, we replace a linear layer with a Multi-Layer Perceptron (MLP) and consider only the mask features, whereas the authors of Interactron [7] use predicted boxes and box features as input to the Prediction Embedder of the Fusion Module.

The rest of the Fusion Module rests as introduced in the original work [7]. Therefore our Fusion Module contains MLP decoders for the learned loss, masks, logits, and actions. The Mask Decoder consists of an MLP that computes the mask embeddings from the Transformer Module outputs. However, in our experiments, we use only the learned loss output.

During training, the parameters ϕ of the Fusion Module are updated by the ground-truth loss computed from the segmentation annotation and predictions made by the segmentation model after the backpropagation of adaptive gradients. Then, the parameters of the segmentation model are optimized to reduce the ground-truth loss with adapted weights. During inference, there is no ground truth, and the parameters of the segmentation model are updated by the learned loss predicted by the Fusion Module.

4. Datasets for adaptive learning in indoor environment

Habitat environment. To train our SegmATRon models, we collected a dataset of 1160 action trees in train scenes of HM3DSem v0.2 [51]. A validation dataset of 144 action trees was collected from validation scenes of HM3DSem v0.2. For the train and the validation datasets, we considered all possible combinations of 4 additional frames obtained with the following agent actions: turn left, turn right, look up, look down, and move backward. The last action corresponds to observing a scene from a more distant point of view. All rotations are made by 30°. Thus, the training set contains 725k possible image sequences corresponding to 112k unique images, whereas the validation set consists of 90k possible image sequences corresponding to 14k unique images. Different sequences of images represent different samples for optimizing the SegmATRon weights.

Table 1
Fusion module component parameters.

Component	Layers	Heads	Hidden Dim
Image feature embedder (Linear)	1	–	–
Prediction embedder (MLP)	3	–	512
Transformer (GPT-2)	4	8	512
Action decoder (MLP)	3	–	512
Mask decoder (MLP)	3	–	512
Logit decoder (Linear)	1	–	–
Loss decoder (MLP)	3	–	512

Since HM3DSem v0.2 contains two sets of categories for semantic segmentation annotation, the first set contains 40 Matterport3D categories [61]. The second set contains a rich semantic with 1624 categories. We decided to leverage this large set of categories and map them into 150 ADE20k [62] categories, which allowed us to get ground truth semantics without pseudo-labeling. For matching categories, we left their original names. Object categories having supercategories in the ADE20K [62] dataset were assigned to their supercategory (e.g., wine bottle - bottle, apple - food, solid food). Small objects with a familiar location in scenes were assigned to their location (e.g., pen-desk). Small objects that do not have a fixed location were categorized as unlabeled (e.g., sponge - unlabeled).

The frame rendering parameters correspond to the Habitat Navigation Challenge 2023 [16] configuration. In particular, the image size was fixed to 640 × 480, horizontal field of view angle was equal to 42°.

AI2-THOR environment. To test the domain adaptation ability of our models, we collected a test dataset of 100 action trees in the test

Table 2
Comparison of SegmATRon method and the state-of-the-art Mask2Former and MaskDINO models on the Habitat dataset with 150 categories. In parentheses, here and below, we show the relative increment of the quality metric compared to the baseline. MD denotes MaskDINO, and M2F denotes Mask2Former.

Method	Adaptation on inference	Action policy	$mIoU$, %	$fwIoU$, %	$mACC$, %	$pACC$, %
MaskDINO	No	Single frame	20.7	64.5	29.2	75.9
SegmATRon (MD) 4 steps	No	Random	18.2 (−12.0%)	60.1 (−6.8%)	25.9 (−11.3%)	72.3 (−4.7%)
SegmATRon (MD) 4 steps	Yes	Random (mean)	20.7 (+0.0%)	65.2 (+1.1%)	28.9 (−1.3%)	76.1 (+0.3%)
SegmATRon (MD) 4 steps	Yes	Random (max)	22.5 (+8.7%)	66.1 (+2.5%)	30.7 (+5.1%)	76.8 (+1.2%)
Mask2Former	No	Single frame	21.9	65.1	31.3	76.6
SegmATRon (M2F) 4 steps	No	Random	20.5 (−6.4%)	62.5 (−4.0%)	27.5 (−12.1%)	75.9 (−0.9%)
SegmATRon (M2F) 4 steps	Yes	Random (mean)	22.6 (+3.2%)	66.5 (+2.2%)	31.6 (+1.0%)	78.0 (+1.8%)
SegmATRon (M2F) 4 steps	Yes	Random (max)	**23.7** (+8.2%)	**67.0** (+2.9%)	**32.7** (+4.5%)	**78.5** (+2.5%)
SegmATRon (M2F) 4 steps	Yes	Best loss	23.3 (+6.4%)	66.9 (+2.8%)	**32.7** (+2.6%)	78.2 (+2.1%)

scenes of the iTHOR synthetic environment [17] using the same set of actions and the same render settings as for the Habitat environment. As the categories set in the AI2-THOR simulator differ from the environment in the Habitat simulator, we considered only 45 intersecting categories from the available 125 categories in the iTHOR scenes. The test set contains 62.5k possible image sequences corresponding to 10k unique images.

Datasets statistics. Fig. 3 presents class distribution statistics for the collected datasets. The histogram shows the distribution of classes in the initial images from the action trees. Given the large number of ADE20k classes (150 categories) spanning both indoor and outdoor spaces, only categories present in the validation set are visualized.

5. Experiments

Training setup. We train neural network models on a server with 2×Nvidia Tesla V100 GPU. The weights of Mask2Former and MaskDINO are initialized by respective models pre-trained on ADE20k [62]. To train SegmATRon as well as Mask2Former and MaskDINO we follow a training procedure described by authors of Interactron [7], but we reduce the epoch number to 120 due to the fast convergence of the segmentation model. We train the models using the AdamW optimizer with $\beta_1 = 0.9, \beta_2 = 0.999$, $\lambda = 0.05$, gradient clipping with a max norm of 0.01 and batch size of 16. The learning rate for the segmentation model was set to 10^{-5}, and the learning rate for the Fusion module was equal to 10^{-4}. For each model design, we run the training process once. During the training process of SegmATRon, we resize input images to 320×240 resolution and pad the image to have a square shape of 320×320.

After training for 120 epochs, we choose checkpoints with the best $mIoU$ value on the validation dataset. We report standard metrics for semantic segmentation [18]: mean Intersection over union ($mIoU$), frequency-weighted Intersection over union ($fwIoU$), mean pixel accuracy ($mACC$) and pixel accuracy ($pACC$).

Implementation details. We provide Table 1 summarizing the parameters of the GPT-based Fusion Module used in the main experiments. For the segmentation models Mask2Former and MaskDINO, we use the same parameters as those provided by the authors for the ADE20k dataset and the ResNet50 backbone.

Single Frame baselines. To distinguish the role of the adaptive learned loss function from the role of fine-tuning the segmentation model, we experimented with fine-tuning the Mask2Former and MaskDINO model (ResNet-50 backbone) without the Fusion Module, following the segmentation model training procedure in the SegmATron architecture.

Results. The SegmATRon with Random rotation action policy significantly outperforms the baseline Mask2Former and MaskDINO approaches (see Table 2) on the validation dataset collected in the Habitat environment in terms of the segmentation quality metrics. Since the SegmATRon approach requires the backpropagation of adaptive gradients during inference, more computing resources are needed for this method. We show that the learned hybrid multicomponent loss function increases the segmentation quality during inference via adaptive gradients. The results from Table 2 demonstrate the crucial role of the adaptive gradients during inference for the SegmATRon approach.

The performance of SegmATRon depends on the sequence of actions chosen by the agent. To confirm the effectiveness of the proposed approach, we conduct 500 validation runs for SegmATRon (Mask2Former) and SegmATRon (MaskDINO) using a random rotation policy. Additionally, we perform 500 runs, sampling actions according to the learned Weighted Best Loss policy for SegmATRon (Mask2Former). We compute the mean and standard deviation of the mIoU, fwIoU, mACC, and pACC metrics on the validation dataset in Habitat for the conducted runs. Table 2 presents the average and maximum metric values across 500 runs.

To confirm a statistically significant improvement in the metrics, we perform a one-sided t-test with the null hypothesis that the mean metric distribution of SegmATRon (Mask2Former/MaskDINO) across runs is less than or equal to the mean metric distribution of the corresponding Single Frame baseline. Additionally, for SegmATRon (Mask2Former), we conduct a t-test with the null hypothesis that the mean metric distribution of SegmATRon Mask2Former with the learned policy across runs is less than or equal to the mean metric distribution of the corresponding SegmATRon Mask2Former with a random policy.

Fig. 5 demonstrates that SegmATRon (Mask2Former) shows a statistically significant improvement in all four metrics compared to the baseline Single Frame approach, even with a random policy. The use of the learned policy allows for an even greater improvement in the segmentation for SegmATRon (Mask2Former). SegmATRon (MaskDINO) (see Fig. 6) with a random policy shows a statistically significant improvement in the frequency-weighted metrics fwIoU and pACC.

Fig. 4 shows the visualized results of SegmATRon compared to Mask2Former and MaskDINO baselines under various scenes from Habitat and AI2-THOR simulators. The SegmATRon models help to correctly predict the object masks located in the corners or along the edges of the first image in a sequence. In the first image, both the SegmATRon (Mask2Former) and the SegmATRon (MaskDINO) correctly discern a cabinet from a chest. In the second image, both the SegmATRon (Mask2Former) and the SegmATRon (MaskDINO) are capable of accurately segmenting paintings on the wall. Moreover, the SegmATRon (MaskDINO) recognizes a chair from an armchair in front of the camera. The SegmATRon (Mask2Former) correctly predicts a mask for the second chair in the room. The third image demonstrates the improvement of sink segmentation.

In the last two images, one can see a black background in the ground truth masks. It is a distinctive characteristic of the data compiled using AI2-THOR, which includes the "background" category. In the fourth image, the SegmATRon (Mask2Former) accurately predicts a mask for an armchair, whereas Mask2Former classifies it as a computer. In the fifth image, our approach correctly predicts the mirror mask.

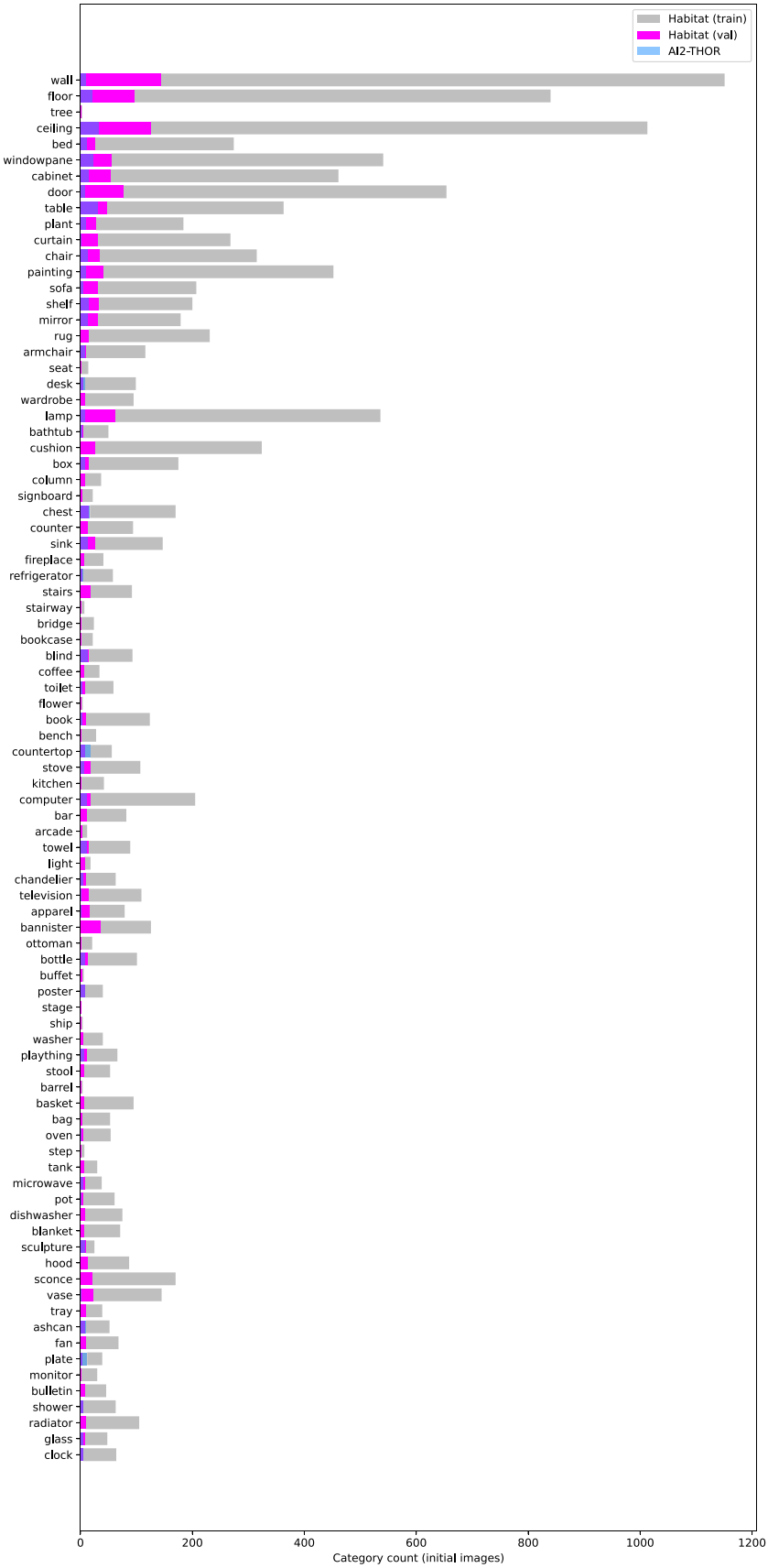


Fig. 3. Histogram illustrating the class distribution from ADE20k across the collected Habitat training and validation datasets, along with the AI2-THOR test dataset. Sample counts for the training, validation, and test sets are depicted in gray, pink, and blue, respectively.

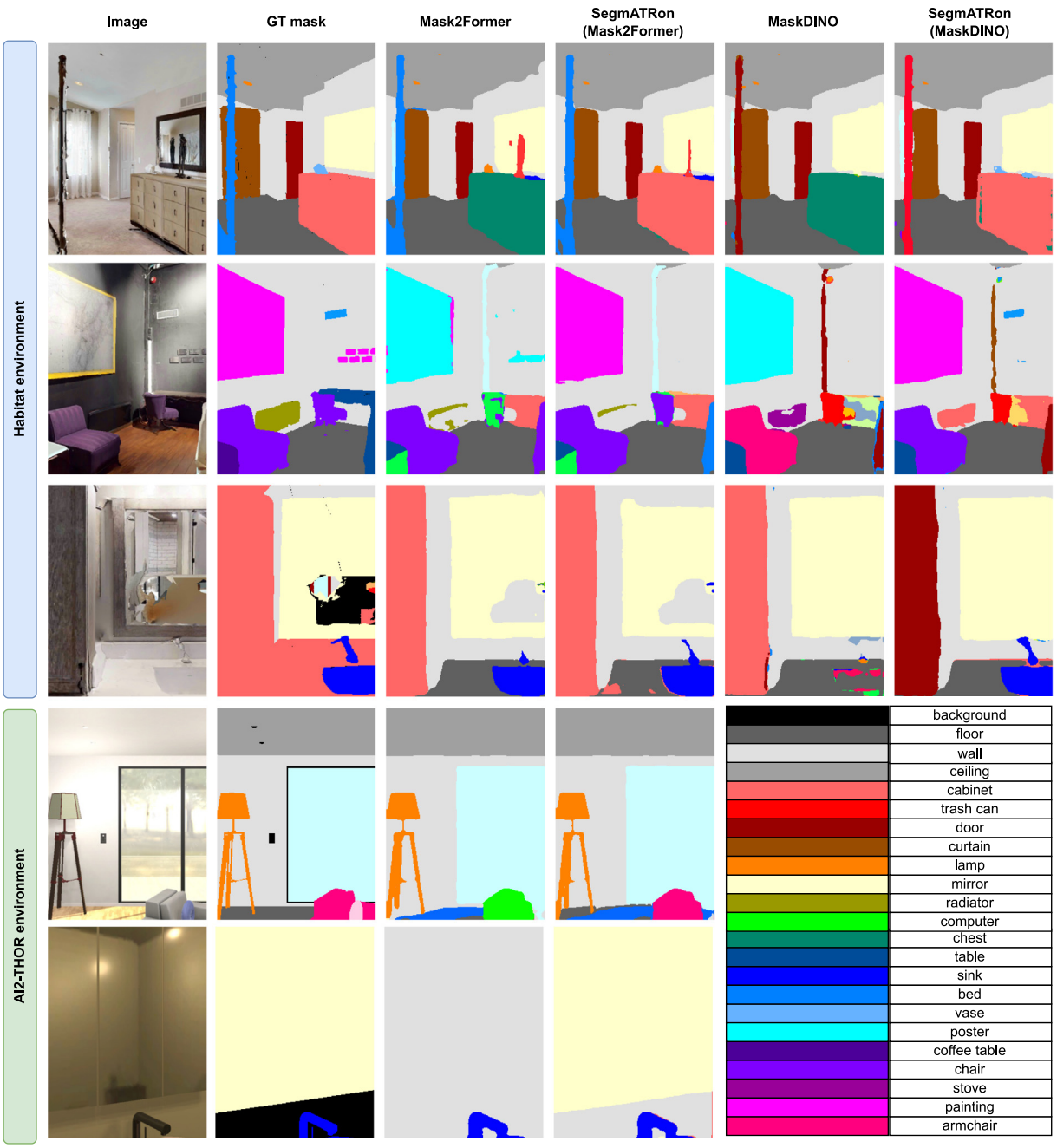


Fig. 4. Visualized segmentation results on Habitat and AI2-THOR validation sets. The columns left-to-right refer to the input image, the ground truth, the outputs of the Mask2Former, the SegmATRon (Mask2Former), the MaskDINO and the SegmATRon (MaskDINO) models. The SegmATRon models improve the segmentation masks for objects located in the corners or along the edges of the first image in a sequence.

Figs. 7 and 8 show the image sequences that the Mask2Former-based SegmATRon uses to improve the quality of the initial image segmentation by the information fusion. The sequences show that rotating towards objects at the image edges improves segmentation quality.

Fig. 9 shows the image sequences that the MaskDINO-based SegmATRon uses to improve the quality of the initial image segmentation by the information fusion. From these sequences, as observed in its results with the Mask2Former-based SegmATRon, it can be seen that rotations towards objects located at the edge of the image improve the quality of their segmentation as well as observing them from additional view points.

Fig. 10 shows more segmentation results of SegmATRon models compared to the baseline models, Mask2Former and MaskDINO, on the images rendered with Habitat. In the provided examples, SegmATRon models more accurately identify the object class compared to baselines and achieve greater precision in delineating object masks.

Fig. 11 shows more segmentation results of SegmATRon models compared to the baseline models on the images rendered with AI2-THOR. Here, as observed in its results with Habitat, SegmATRon frequently exhibits more accurate classification of segmented objects compared to the baseline models.

6. Ablation studies

We analyze SegmATRon's components through a series of ablation studies.

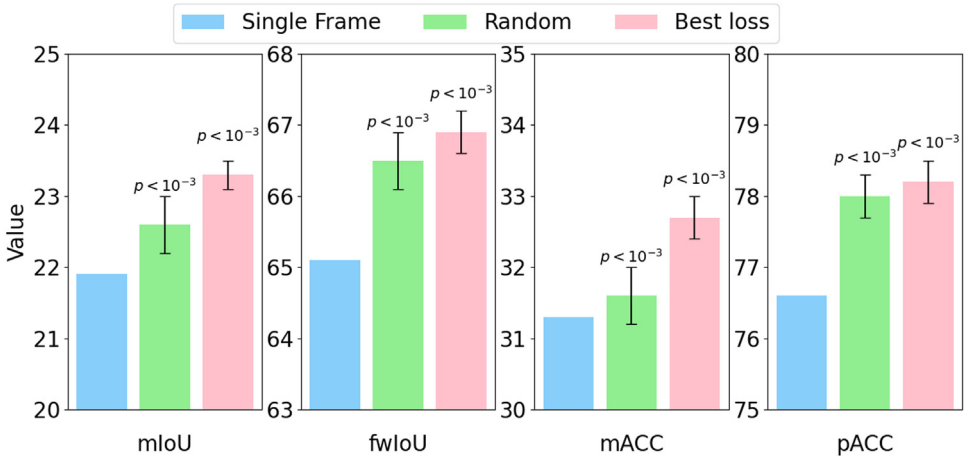


Fig. 5. Comparison of baseline Single Frame Mask2Former, SegmATRon (Mask2Former) with Random rotation and Best loss action policies across four evaluation metrics (mIoU, fwIoU, mACC, pACC). Error bars represent standard deviations, and statistical significance is indicated by p-values $p < 10^{-3}$, demonstrating significant improvements of the SegmATRon (Mask2Former) approach over the Single Frame baseline.

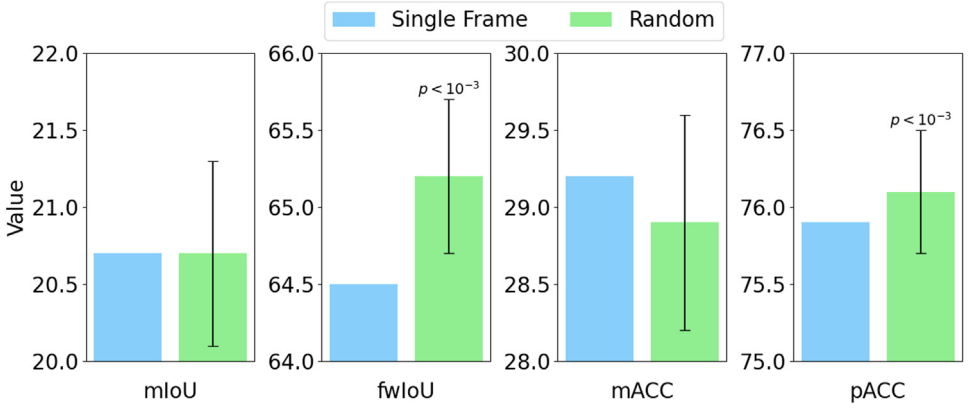


Fig. 6. Comparison of baseline Single Frame MaskDINO and SegmATRon (MaskDINO) with Random rotation action policy across four evaluation metrics (mIoU, fwIoU, mACC, pACC). Error bars represent standard deviations, and statistical significance is indicated by $p < 10^{-3}$, demonstrating significant improvements of the SegmATRon (MaskDINO) approach over the Single Frame baseline for frequency-weighted metrics $fwIoU$ and $pACC$.

Table 3
Ablation study. The number of steps (additional frames). N_p denotes the number of parameters in neural network model. Inference speed is measured on NVIDIA GeForce RTX 3060.

Method	Adaptation on inference	Steps	$mIoU$, %	$fwIoU$, %	$mACC$, %	$pACC$, %	N_p	FPS	VRAM, Gb
Mask2Former	–	–	21.9	65.1	31.3	76.6	44M	29.4	2.9
Mask2Former	Yes	–	22.6 (+3.1%)	66.2 (+1.7%)	32.1 (+2.6%)	77.8 (+1.6%)	60M	8.8	3.5
SegmATRon (M2F)	Yes	1	23.4 (+6.8%)	65.0 (−0.2%)	**33.6** (+7.3%)	76.6 (+0.0%)	60M	7.3	4.0
SegmATRon (M2F)	Yes	2	**24.0** (+9.6%)	66.1 (+1.5%)	32.5 (+3.8%)	77.6 (+1.3%)	60M	5.9	4.8
SegmATRon (M2F)	Yes	3	23.7 (+8.2%)	66.3 (+1.8%)	32.2 (+2.9%)	77.9 (+1.7%)	60M	4.8	5.6
SegmATRon (M2F)	Yes	4	23.7 (+8.2%)	**67.0** (+2.9%)	32.7 (+4.5%)	**78.5** (+2.5%)	60M	4.0	6.8

Number of Steps (Additional Frames). We study the influence of the frame number used for the prediction of the learned loss function. For each number of additional steps N, we train a version of the SegmATRon model using N frames during training. As one can see from Table 3, the use of 4 additional frames instead of 1, 2, 3 improves the performance of the SegmATRon (Mask2Former) model considering the metrics $fwIoU$ and $pACC$. All other models excel in only one type of metric. Additional frames utilization does not drastically increase inference time compared to the Mask2Former model with adaptive loss, but significantly improves segmentation quality. We measure FPS (frames per second) during inference with $N = \{1, 2, 3, 4\}$ pre-collected frames as the model input. That is, we do not consider the time for predicting the next action to collect a frame, since it can be different depending on the choice of the policy type.

We observe that despite the improvement in segmentation quality with an increasing number of frames in the sequence, the amount of GPU memory required for model inference also increases. With a further increase in the number of steps, we expect a significant rise in GPU memory consumption. We propose SegmATRon as a segmentation method for a mobile intelligent agent. Therefore, we did not consider further increasing the number of frames in the sequence since it is impractical due to the limited GPU memory available on the robot's onboard computers.

The decrease in SegmATRon (Mask2Former) inference speed is due to the following additional operations. First, the segmentation

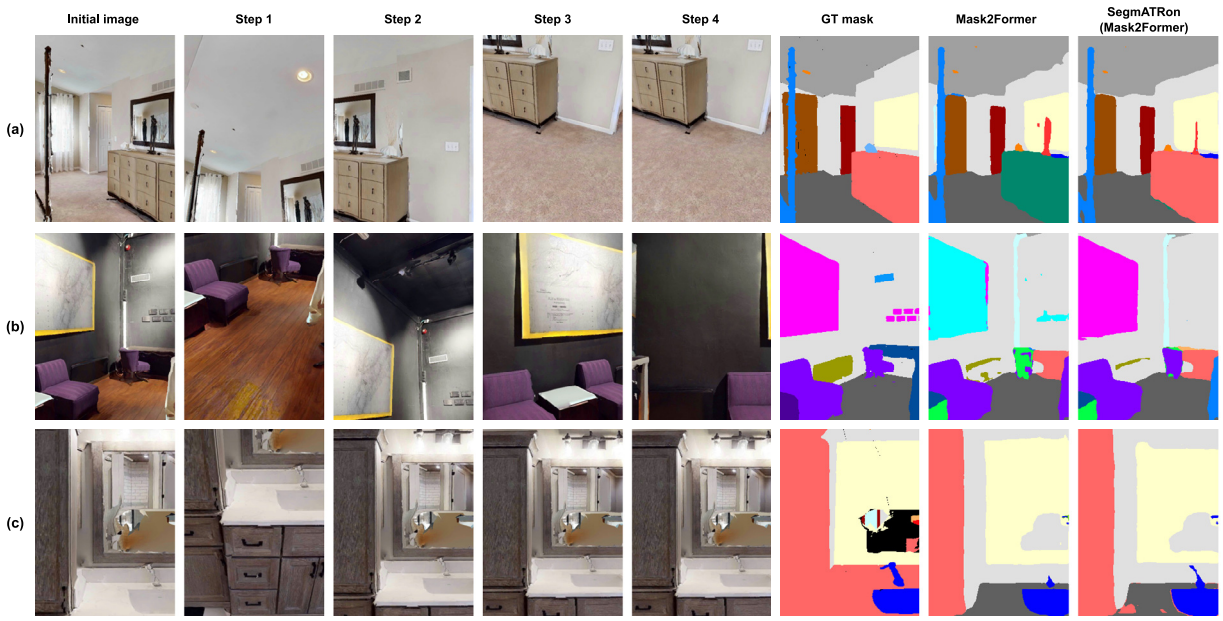


Fig. 7. Visualized segmentation results of the Mask2Former-based SegmATRon on the Habitat validation set. The columns left-to-right refer to the input image, the collected additional images, the ground truth, the outputs of the Mask2Former model and the outputs of the Mask2Former-based SegmATRon. (a) SegmATRon (Mask2Former) utilizes additional information from the scene's new frames to accurately classify the object as a cabinet, whereas Mask2Former misidentifies it as a chest. (b) SegmATRon (Mask2Former) correctly identifies the painting and chair located in the corner of the room, while Mask2Former erroneously labels these objects as a poster and a computer, respectively. (c) SegmATRon (Mask2Former) successfully improves the sink segmentation mask compared to Mask2Former.

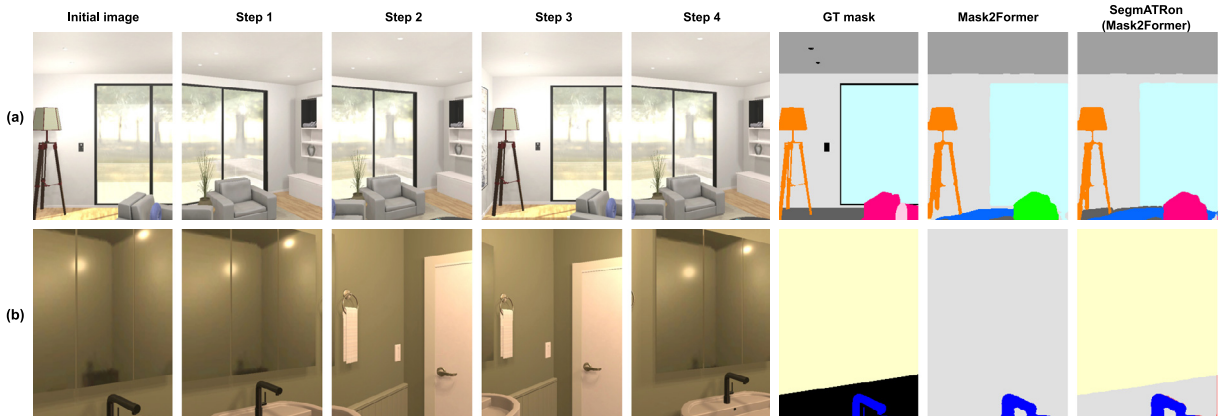


Fig. 8. Visualized segmentation results of the Mask2Former-based SegmATRon on the AI2-THOR test set. The columns left-to-right refer to the input image, the collected additional images, the ground truth, the outputs of the Mask2Former model and the outputs of the Mask2Former-based SegmATRon. (a) SegmATRon (Mask2Former) accurately classify the armchair, whereas Mask2Former misidentifies it as a computer. (b) SegmATRon (Mask2Former) successfully identifies the mirror, whereas Mask2Former fails to segment it.

model performs inference on additional frames while preserving the computation graph for subsequent gradient computation. Second, the Fusion Module predicts learned adaptive loss function value. After that, adaptive gradients are computed with respect to the predicted adaptive loss function. Finally, the segmentation model is called again with updated weights for the first frame.

Table 4 summarizes the time required for each stage of Mask2Former (SegmATRon) inference. It is important to note that while performing segmentation model inference on additional frames increases computational complexity, it can be parallelized.

Fusion Module Architecture. We explore the impact of the transformer architecture choice in the Fusion module. In these experiments, we use the same number of layers, heads, and hidden dimensions for both transformers. Table 5 shows that using the GPT transformer to merge information from different frames provides a significant advantage over the DETR transformer architecture used in the [7] paper. The DETR transformer requires fewer FLOPs than GPT, which affects the inference speed of the Fusion Module. However, in terms of inference time, both transformers are faster compared to the segmentation model. Therefore, we choose GPT for further experiments. In addition, the GPT architecture can easily handle sequences of frames of varying lengths, allowing it to be used for action prediction for collecting the next frame.

Domain Adaptation. We demonstrate the benefit of an adaptive loss function when changing the photorealistic Habitat domain to the synthetic AI2-THOR images. We run inference on the SegmATRon (Mask2Former) and SegmATRon (MaskDINO) models, which were trained on a dataset from the Habitat environment, using a test dataset we collected in AI2-THOR. Table 6 demonstrates the advantage of the adaptive loss function for the SegmATRon (Mask2Former) model. The SegmATRon (Mask2Former) is capable of adapting efficiently to a new type of environment compared to the single-frame Mask2Former. For the SegmATRon (MaskDINO) model, it is worth noting the positive

Fig. 9. Visualized segmentation results of the MaskDINO-based SegmATRon on the Habitat validation set. The columns left-to-right refer to the input image, the collected additional images, the ground truth, the outputs of the MaskDINO model and the outputs of the MaskDINO-based SegmATRon. (a) SegmATRon (MaskDINO) accurately identifies the cabinet, while MaskDINO misclassifies it as a chest. (b) The use of additional frames enables SegmATRon (MaskDINO) to correctly classify the painting, whereas MaskDINO incorrectly predicts it as a poster. (c) Similar to Fig. 7, SegmATRon (MaskDINO) successfully detects the sink, while MaskDINO fails to segment it.

Table 4
Performance comparison of different elements of SegmATRon (Mask2Former) inference. Inference speed is measured on NVIDIA GeForce RTX 3060.

Operation	Steps	Inference time, ms	FLOPs
Inference of Mask2Former	–	34	132B
Inference of Mask2Former (before adaptation on inference)	1	53	264B
Inference of Fusion Module	1	4	36B
Adaptive Gradients computation	1	46	52B
Inference of Mask2Former (after adaptation on inference)	–	34	132B
Total	1	137	484B
Inference of Mask2Former (before adaptation on inference)	2	67	477B
Inference of Fusion Module	2	6	56B
Adaptive Gradients computation	2	62	79B
Inference of Mask2Former (after adaptation on inference)	–	34	132B
Total	2	169	744B
Inference of Mask2Former (before adaptation on inference)	3	84	636B
Inference of Fusion Module	3	9	78B
Adaptive Gradients computation	3	82	107B
Inference of Mask2Former (after adaptation on inference)	–	34	132B
Total	3	209	953B
Inference of Mask2Former (before adaptation on inference)	4	100	794B
Inference of Fusion Module	4	12	101B
Adaptive Gradients computation	4	103	135B
Inference of Mask2Former (after adaptation on inference)	–	34	132B
Total	4	249	1162B

Table 5
Ablation study. Fusion module transformer. For comparison, we provide information on the number of parameters N_p, FPS, and FLOPs for the Fusion Module transformer.

Method	Fusion Module	$mIoU$, %	$fwIoU$, %	$mACC$, %	$pACC$, %	N_p	FPS	FLOPs
Mask2Former	–	21.9	65.1	31.3	76.6			
SegmATRon (M2F) 4 steps	DETR	22.4 (+2.3%)	64.8 (−0.5%)	**32.1** (+2.6%)	76.1 (−0.7%)	23M	**138**	**48B**
SegmATRon (M2F) 4 steps	GPT	**23.7** (+8.2%)	**67.0** (+2.9%)	**32.7** (+4.5%)	**78.5** (+2.5%)	**17M**	82	101B

effect of including the adaptive loss function on the inference compared to the version of SegmATRon (MaskDINO) without adapting the weights on the inference.

Policy optimization. Finally, we study the optimization of the policy of choosing the next frame in the sequence. We adopt the approach proposed by the authors [7]. As one can see from Table 7, this

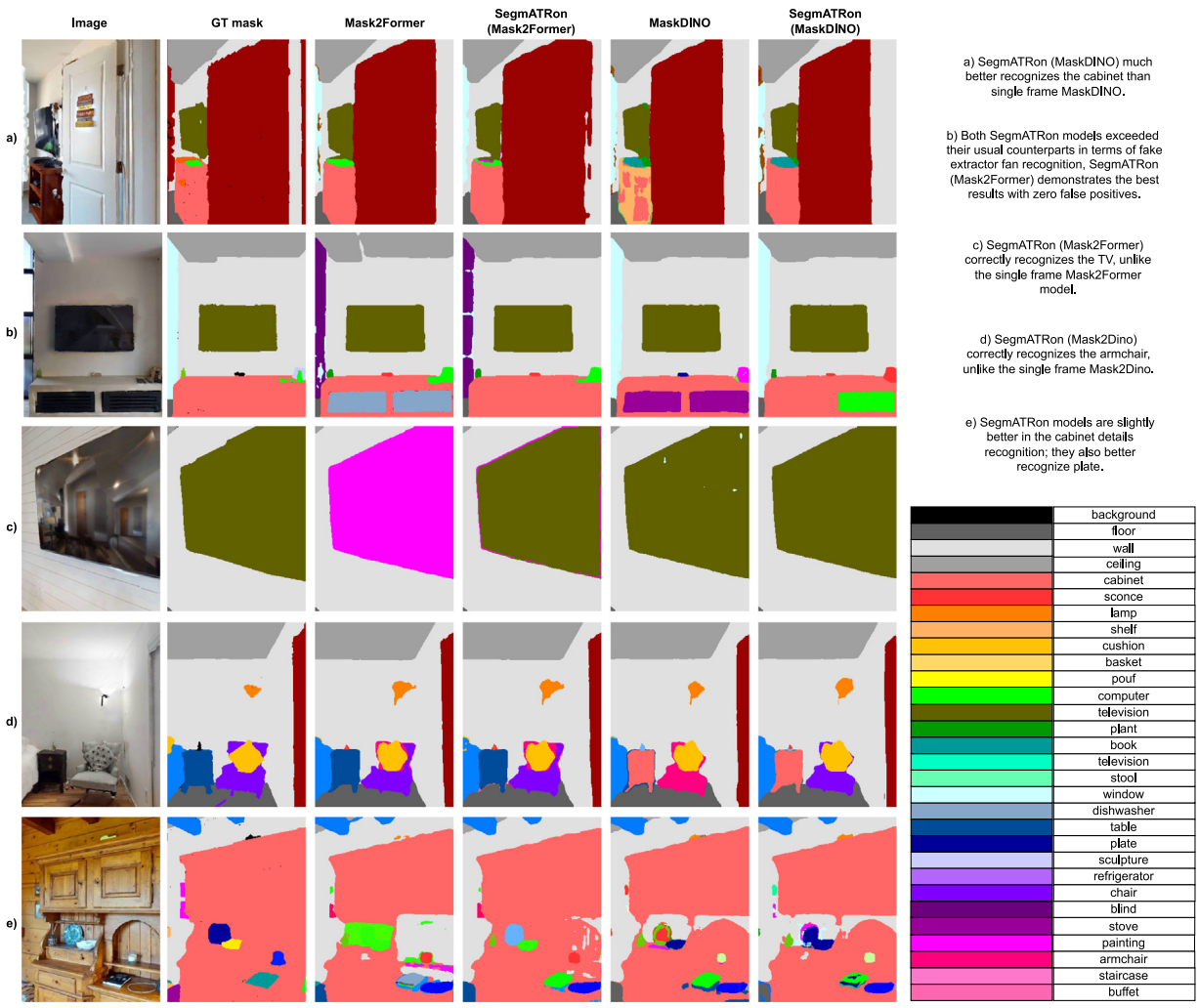


Fig. 10. Visualized segmentation results on the Habitat validation set. The columns left-to-right refer to the input image, the ground truth, the outputs of the Mask2Former model, the outputs of the Mask2Former-based SegmATRon, MaskDINO outputs and MaskDINO-based SegmATRon results.

Table 6
Ablation study. Domain adaptation on the AI2-THOR dataset.

Method	Adaptation on inference	Action policy	Train. dataset	Val. dataset	$mIoU$, %	$fwIoU$, %	$mACC$, %	$pACC$, %
Mask2Former	No	Single frame	Habitat	AI2-THOR	22.3	40.4	30.1	54.0
SegmATRon (M2F) 4 steps	No	Random	Habitat	AI2-THOR	18.2 (−18.4%)	39.0 (−2.5%)	24.5 (−18.6%)	52.9 (−2.0%)
SegmATRon (M2F) 4 steps	Yes	Random	Habitat	AI2-THOR	22.8 (+2.2%)	46.3 (+14.6%)	31.5 (+4.7%)	60.6 (+12.2%)
MaskDINO	No	Single frame	Habitat	AI2-THOR	**34.3**	**53.1**	**45.1**	**65.9**
SegmATRon (MD) 4 steps	No	Random	Habitat	AI2-THOR	23.0 (−32.9%)	43.6 (−17.9%)	29.9 (−33.7%)	58.1 (−11.8%)
SegmATRon (MD) 4 steps	Yes	Random	Habitat	AI2-THOR	27.1 (−20.1%)	50.0 (−5.8%)	36.9 (−18.2%)	63.4 (−3.8%)

Table 7
Ablation study. Policy optimization.

Method	Adaptation on inference	Action policy	$mIoU$, %	$fwIoU$, %	$mACC$, %	$pACC$, %
Mask2Former	No	Single frame	21.9	65.1	31.3	76.6
SegmATRon (M2F) 4 steps	Yes	Random (mean)	22.6 ± 0.4	66.5 ± 0.4	31.6 ± 0.4	78.0 ± 0.3
SegmATRon (M2F) 4 steps	Yes	Weighted Best loss	**23.3**	**66.9**	**32.7**	**78.2**

approach for policy optimization improves the segmentation quality of SegmATRon (Mask2Former) in terms of $mIoU$ and $mACC$ metrics. Fig. 12 shows the qualitative difference between the random actions selection and the Weighted Best Loss policy.

7. Possible applications

Domain Adaption. SegmATRon demonstrates a higher ability to adapt to new domains compared to the Single Frame baselines, as

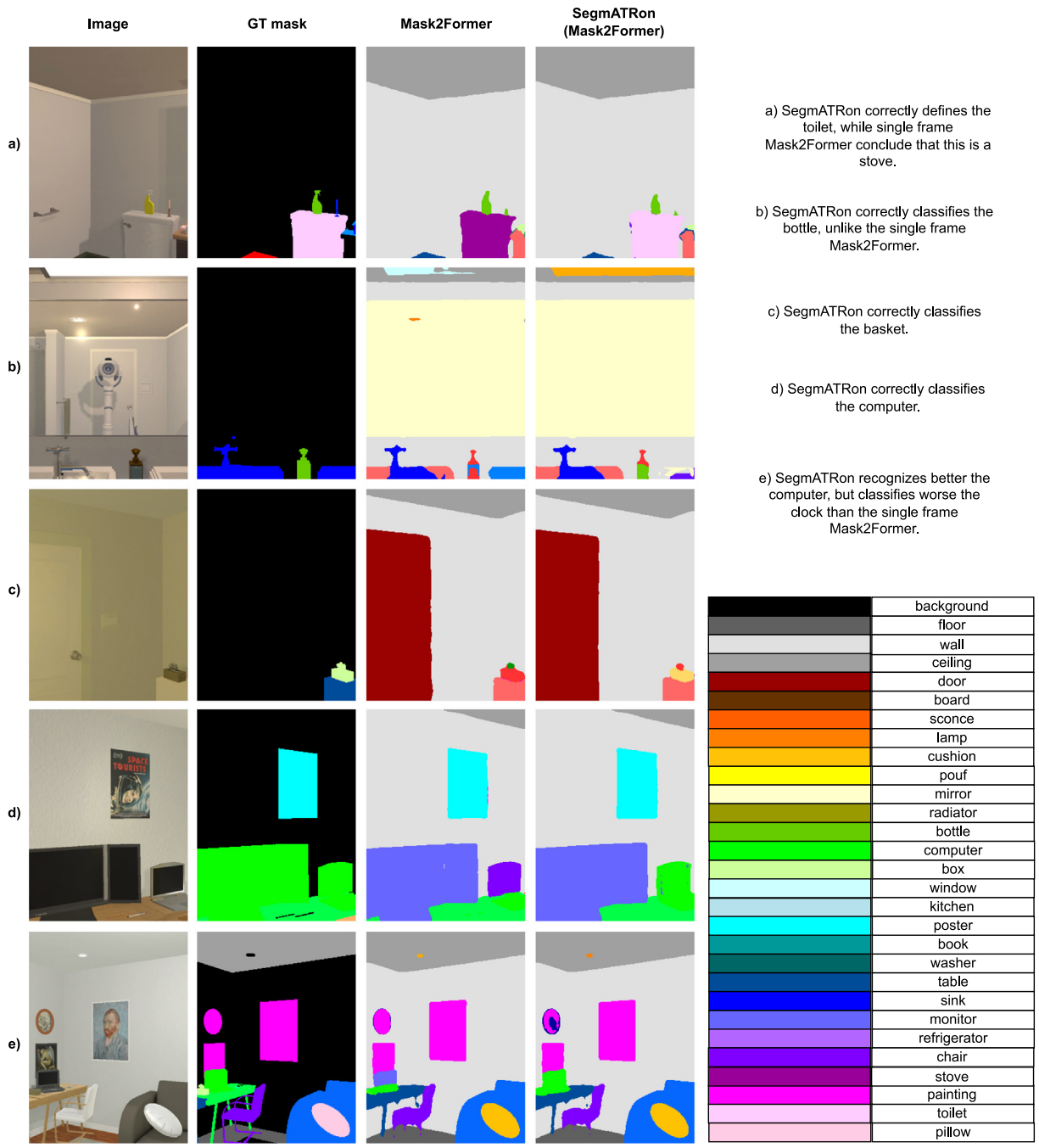


Fig. 11. Visualized segmentation results on the AI2-THOR validation set. The columns left-to-right refer to the input image, the ground truth, the outputs of the Mask2Former model and the outputs of the Mask2Former-based SegmATRon.

shown by the experiments in Table 6. Therefore, SegmATRon can be used to adapt a semantic segmentation model to new domains without additional fine-tuning if the agent can collect frame sequences in the new environment during inference.

Robot navigation with decoupled navigation policy. One possible real-world application of SegmATRon would be its use in visual indoor navigation. If an indoor navigation method relies on the semantics of the surrounding environment to predict an action policy in the form of an instantaneous or accumulated semantic map, the quality of navigation will depend on the accuracy of the semantic map construction. Our experiments show that SegmATRon can be used to improve segmentation quality under random rotations. Thus, SegmATRon can be paired with a navigation policy, where its observations will be used to form sequences of frames over which SegmATRon will aggregate information.

Robot navigation with learned policy for active perception. Additionally, SegmATRon, with a learned policy for collecting new frames in a sequence, can be used alongside the primary navigation policy to further improve segmentation quality and semantic map construction.

Resource consumption. Real-world application scenarios impose certain requirements on the model's computational efficiency and performance speed. As shown in Table 3, SegmATRon demonstrates near real-time performance with 1 additional frame. During inference of all versions of the model, video memory consumption remains below 8 GB, which aligns with the specifications of modern on-board GPUs, such as the Nvidia GeForce RTX3080 16 GB Mobile.

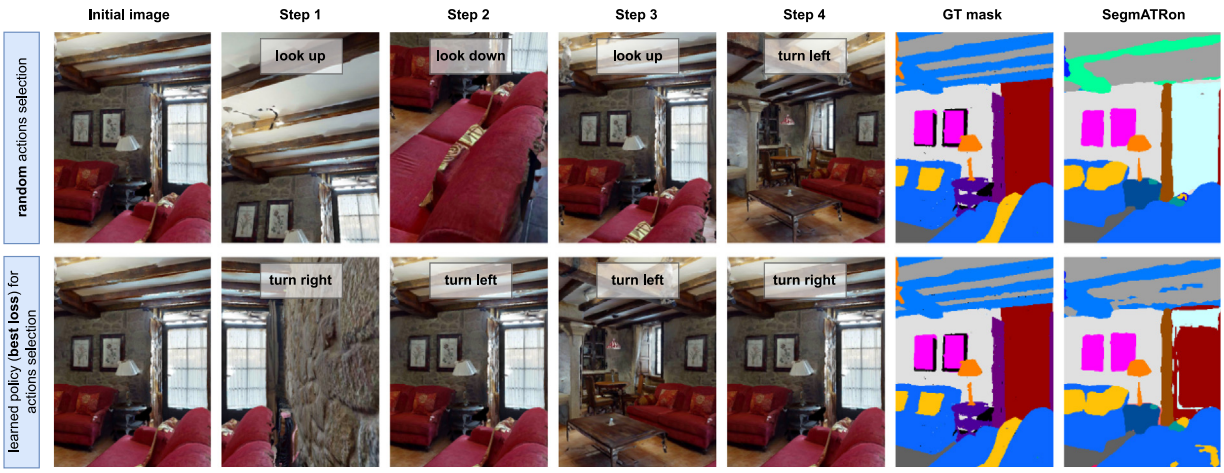


Fig. 12. Visualized segmentation results on the Habitat validation set. The columns left-to-right refer to the input image, the collected additional images, the ground truth and the outputs of the Mask2Former-based SegmATRon. SegmATRon (Weighted Best Loss) leverages rotational views to gather additional scene information, enabling accurate identification of the bannister and the door, whereas SegmATRon (Random) misclassifies them as a bar and a window, respectively.

Table 8
An example of accelerating SegmATRon (Mask2Former) inference by reducing the frequency of gradient computations for the adaptive loss function. Inference speed is measured on NVIDIA GeForce RTX 3060.

Method	Adaptation on inference	Steps	$mIoU$, %	$fwIoU$, %	$mACC$, %	$pACC$, %	N_p	FPS
Mask2Former	–	–	21.0	65.0	30.1	76.7	44M	29.4
SegmATRon (M2F)	Yes, every 5 steps	4	**21.4** (+1.9%)	**65.4** (+0.6%)	**30.1** (+0.0%)	**77.1** (+0.5%)	60M	12.4

Inference acceleration. During training, SegmATRon learns to predict the adaptive loss for all frame sequences contained in the training dataset. Performance analysis of SegmATRon also shows that gradient computations for weight adaptation are a computationally expensive operation. Therefore, for real-world deployment of SegmATRon in a navigation task, the number of weight adaptation calls during inference can be reduced by performing one weight adaptation for every five frames. The results in Table 8 show that, under this setup, SegmATRon achieves an inference speed of 12 FPS while maintaining a segmentation quality advantage over the Single Frame baseline. In this experiment, we use the Random Action policy corresponding to the scenario where actions are sampled with a navigation policy.

8. Conclusion

Our results show that the semantic segmentation quality benefits from the mechanism of multicomponent loss learning which allows us to use additional points of view. We have also demonstrated that the action strategy has a noticeable impact on the result, while further research on the number of actions and their automatic learning is reasonable. The proposed approach is valuable for a navigation task in the environment, where the agent can use the observation history to improve the quality of segmentation for the current frame.

As a limitation of the proposed approach, we can highlight the difficulty of scaling the approach to more than 4 steps. In this case, the need for video memory increases significantly. Another limitation is the small number of existing datasets for training and testing embodied segmentation methods.

A future perspective for the SegmATRon approach would be action policy optimization via Reinforcement Learning based on segmentation loss, which we are currently working on. Other promising future directions are the study of other basic semantic segmentation models as part of the proposed approach, as well as its application to solve the problem of instance segmentation.

CRediT authorship contribution statement

Tatiana Zemskova: Writing – original draft, Validation, Software, Methodology, Investigation, Data curation. **Margarita Kichik:** Writing – original draft, Visualization, Validation, Software, Investigation. **Dmitry Yudin:** Writing – review & editing, Writing – original draft, Visualization, Supervision, Project administration, Funding acquisition, Conceptualization. **Aleksei Staroverov:** Writing – review & editing, Software, Data curation. **Aleksandr Panov:** Writing – review & editing, Supervision, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by Russian Science Foundation, grant No. 20-71-10116, https://rscf.ru/en/project/20-71-10116/.

Data availability

The code of the proposed approach and datasets are publicly available at https://github.com/wingrune/SegmATRon.

References

[1] R. Pfeifer, F. Iida, Embodied artificial intelligence: Trends and challenges, Lecture Notes in Comput. Sci. (2004) 1–26.

[2] M. Deitke, D. Batra, Y. Bisk, T. Campari, A.X. Chang, D.S. Chaplot, C. Chen, C.P. D'Arpino, K. Ehsani, A. Farhadi, et al., Retrospectives on the embodied ai workshop, 2022, arXiv preprint arXiv:2210.06849.

[3] L. Weihs, M. Deitke, A. Kembhavi, R. Mottaghi, Visual room rearrangement, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2021.

[4] R. Partsey, E. Wijmans, N. Yokoyama, O. Dobosevych, D. Batra, O. Maksymets, Is mapping necessary for realistic PointGoal navigation? in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 17232–17241.

[5] A. Staroverov, K. Muravyev, K. Yakovlev, A.I. Panov, Skill fusion in hybrid robotic framework for visual object goal navigation, Robotics 12 (4) (2023) 104.

[6] J. Yang, Z. Ren, M. Xu, X. Chen, D.J. Crandall, D. Parikh, D. Batra, Embodied amodal recognition: Learning to move to perceive objects, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 2040–2050.

[7] K. Kotar, R. Mottaghi, Interactron: Embodied adaptive object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 14860–14869.

[8] Z. Wu, Z. Wang, Z. Wei, Y. Wei, H. Yan, Smart explorer: Recognizing objects in dense clutter via interactive exploration, in: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2022, pp. 6600–6607.

[9] W. Ding, N. Majcherczyk, M. Deshpande, X. Qi, D. Zhao, R. Madhivanan, A. Sen, Learning to view: Decision transformers for active object detection, 2023, arXiv preprint arXiv:2301.09544.

[10] O. Maksymets, V. Cartillier, A. Gokaslan, E. Wijmans, W. Galuba, S. Lee, D. Batra, Thda: Treasure hunt data augmentation for semantic navigation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 15374–15383.

[11] R. Zurbrügg, H. Blum, C. Cadena, R. Siegwart, L. Schmid, Embodied active domain adaptation for semantic segmentation via informative path planning, IEEE Robot. Autom. Lett. 7 (4) (2022) 8691–8698.

[12] D. Nilsson, A. Pirinen, E. Gärtner, C. Sminchisescu, Embodied visual active learning for semantic segmentation, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 2373–2383.

[13] B. Xie, L. Yuan, S. Li, C.H. Liu, X. Cheng, Towards fewer annotations: Active learning via region impurity and prediction uncertainty for domain adaptive semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 8068–8078.

[14] S. Agarwal, S. Anand, C. Arora, Reducing annotation effort by identifying and labeling contextually diverse classes for semantic segmentation under domain shift, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2023, pp. 5904–5913.

[15] M. Wortsman, K. Ehsani, M. Rastegari, A. Farhadi, R. Mottaghi, Learning to learn how to learn: Self-adaptive visual navigation using meta-learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 6750–6759.

[16] 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, D. Batra, Habitat challenge 2023, 2023.

[17] E. Kolve, R. Mottaghi, W. Han, E. VanderBilt, L. Weihs, A. Herrasti, M. Deitke, K. Ehsani, D. Gordon, Y. Zhu, et al., Ai2-thor: An interactive 3d environment for visual ai, 2017, arXiv preprint arXiv:1712.05474.

[18] B. Cheng, I. Misra, A.G. Schwing, A. Kirillov, R. Girdhar, Masked-attention mask transformer for universal image segmentation, 2022.

[19] F. Li, H. Zhang, H. Xu, S. Liu, L. Zhang, L.M. Ni, H.-Y. Shum, Mask dino: Towards a unified transformer-based framework for object detection and segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 3041–3050.

[20] W. Wang, J. Dai, Z. Chen, Z. Huang, Z. Li, X. Zhu, X. Hu, T. Lu, L. Lu, H. Li, et al., InternImage: Exploring large-scale vision foundation models with deformable convolutions, 2022, arXiv preprint arXiv:2211.05778.

[21] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, W. Liu, B. Xiao, Deep high-resolution representation learning for visual recognition, TPAMI (2019).

[22] A. Tao, K. Sapra, B. Catanzaro, Hierarchical multi-scale attention for semantic segmentation, 2020, arXiv preprint arXiv:2005.10821.

[23] H. Liu, F. Liu, X. Fan, D. Huang, Polarized self-attention: Towards high-quality pixel-wise regression, 2021, arXiv preprint arXiv:2107.00782.

[24] J. Jain, J. Li, M.T. Chiu, A. Hassani, N. Orlov, H. Shi, Oneformer: One transformer to rule universal image segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 2989–2998.

[25] Q. Yu, H. Wang, S. Qiao, M. Collins, Y. Zhu, H. Adam, A. Yuille, L.-C. Chen, K-means mask transformer, in: European Conference on Computer Vision, Springer, 2022, pp. 288–307.

[26] B. Cheng, M.D. Collins, Y. Zhu, T. Liu, T.S. Huang, H. Adam, L.-C. Chen, Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 12475–12485.

[27] J. Liang, T. Zhou, D. Liu, W. Wang, Clustseg: Clustering for universal segmentation, 2023, arXiv preprint arXiv:2305.02187.

[28] T. Zhou, W. Wang, Prototype-based semantic segmentation, IEEE Trans. Pattern Anal. Mach. Intell. (2024).

[29] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A.C. Berg, W.-Y. Lo, et al., Segment anything, 2023, arXiv preprint arXiv:2304.02643.

[30] M. Liao, G. Hua, S. Tian, Y. Zhang, W. Zou, X. Li, Exploring more concentrated and consistent activation regions for cross-domain semantic segmentation, Neurocomputing 500 (2022) 938–948.

[31] Y. Zhang, S. Tian, M. Liao, W. Zou, C. Xu, A hybrid domain learning framework for unsupervised semantic segmentation, Neurocomputing 516 (2023) 133–145.

[32] Y. Zhang, S. Tian, M. Liao, G. Hua, W. Zou, C. Xu, A global reweighting approach for cross-domain semantic segmentation, Signal Process., Image Commun. 130 (2025) 117197.

[33] T. Zhou, W. Wang, Cross-image pixel contrasting for semantic segmentation, IEEE Trans. Pattern Anal. Mach. Intell. (2024).

[34] W. Zhou, X. Lin, J. Lei, L. Yu, J.-N. Hwang, MFFENet: Multiscale feature fusion and enhancement network for RGB–thermal urban road scene parsing, IEEE Trans. Multimed. 24 (2021) 2526–2538.

[35] W. Zhou, E. Yang, J. Lei, L. Yu, FRNet: Feature reconstruction network for RGB-D indoor scene parsing, IEEE J. Sel. Top. Signal Process. 16 (4) (2022) 677–687.

[36] W. Zhou, S. Dong, J. Lei, L. Yu, MTANet: Multitask-aware network with hierarchical multimodal fusion for RGB-T urban scene understanding, IEEE Trans. Intell. Veh. 8 (1) (2022) 48–58.

[37] W. Zhou, Y. Zhu, J. Lei, R. Yang, L. Yu, LSNet: Lightweight spatial boosting network for detecting salient objects in RGB-thermal images, IEEE Trans. Image Process. 32 (2023) 1329–1340.

[38] W. Zhou, F. Sun, Q. Jiang, R. Cong, J.-N. Hwang, WaveNet: Wavelet network with knowledge distillation for RGB-T salient object detection, IEEE Trans. Image Process. 32 (2023) 3027–3039.

[39] W. Zhou, E. Yang, J. Lei, J. Wan, L. Yu, PGDENet: Progressive guided fusion and depth enhancement network for RGB-D indoor scene parsing, IEEE Trans. Multimed. 25 (2022) 3483–3494.

[40] W. Zhou, G. Xu, M. Fang, S. Mao, R. Yang, L. Yu, PGGNet: Pyramid gradual-guidance network for RGB-D indoor scene semantic segmentation, Signal Process., Image Commun. 128 (2024) 117164.

[41] W. Zhou, Y. Yue, M. Fang, X. Qian, R. Yang, L. Yu, BCINet: Bilateral cross-modal interaction network for indoor scene understanding in RGB-D images, Inf. Fusion 94 (2023) 32–42.

[42] D. Kim, S. Woo, J.-Y. Lee, I.S. Kweon, Video panoptic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2020.

[43] J. Miao, X. Wang, Y. Wu, W. Li, X. Zhang, Y. Wei, Y. Yang, Large-scale video panoptic segmentation in the wild: A benchmark, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 21033–21043.

[44] Y. Xu, Z. Yang, Y. Yang, Video object segmentation in panoptic wild scenes, 2023, arXiv preprint arXiv:2305.04470.

[45] A. Athar, A. Hermans, J. Luiten, D. Ramanan, B. Leibe, Tarvis: A unified approach for target-based video segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 18738–18748.

[46] L. Ke, M. Danelljan, H. Ding, Y.-W. Tai, C.-K. Tang, F. Yu, Mask-free video instance segmentation, in: CVPR, 2023.

[47] T. Zhang, X. Tian, Y. Wu, S. Ji, X. Wang, Y. Zhang, P. Wan, DVIS: Decoupled video instance segmentation framework, 2023, arXiv preprint arXiv:2306.03413.

[48] I. Shin, D. Kim, Q. Yu, J. Xie, H.-S. Kim, B. Green, I.S. Kweon, K.-J. Yoon, L.-C. Chen, Video-kMaX: A simple unified approach for online and near-online video panoptic segmentation, 2023, arXiv preprint arXiv:2304.04694.

[49] L. Li, W. Wang, T. Zhou, J. Li, Y. Yang, Unified mask embedding and correspondence learning for self-supervised video segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 18706–18716.

[50] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, et al., Sam 2: Segment anything in images and videos, 2024, arXiv preprint arXiv:2408.00714.

[51] K. Yadav, R. Ramrakhya, S.K. Ramakrishnan, T. Gervet, J. Turner, A. Gokaslan, N. Maestre, A.X. Chang, D. Batra, M. Savva, et al., Habitat-matterport 3d semantics dataset, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 4927–4936.

[52] G. Chaudhary, L. Behera, T. Sandhan, Active perception system for enhanced visual signal recovery using deep reinforcement learning, in: ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, IEEE, 2023, pp. 1–5.

[53] P. Hoseini, S.K. Paul, M. Nicolescu, M. Nicolescu, A one-shot next best view system for active object recognition, Appl. Intell. 52 (5) (2022) 5290–5309.

[54] Z. Liu, Z. Wang, S. Huang, J. Zhou, J. Lu, GE-grasp: Efficient target-oriented grasping in dense clutter, in: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2022, pp. 1388–1395.

[55] H. Luo, Z. Wu, H. Yan, AE-reorient: Active exploration based reorientation for robotic pick-and-place, 2023.
[56] K.P. Singh, L. Weihs, A. Herrasti, A. Kembhavi, R. Mottaghi, Ask4Help: Learning to leverage an expert for embodied tasks, in: NeurIPS, 2022.
[57] Z. Fang, A. Jain, G. Sarch, A.W. Harley, K. Fragkiadaki, Move to see better: Towards self-supervised amodal object detection, 2020, arXiv:2012.00057.
[58] D.S. Chaplot, M. Dalal, S. Gupta, J. Malik, R.R. Salakhutdinov, Seal: Self-supervised embodied active learning using exploration and 3d consistency, Adv. Neural Inf. Process. Syst. 34 (2021) 13086–13098.
[59] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al., Language models are unsupervised multitask learners, OpenAI Blog 1 (8) (2019) 9.
[60] K. Gupta, T. Ajanthan, A.v.d. Hengel, S. Gould, Understanding and improving the role of projection head in self-supervised learning, 2022, arXiv preprint arXiv:2212.11491.
[61] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niessner, M. Savva, S. Song, A. Zeng, Y. Zhang, Matterport3D: Learning from RGB-D data in indoor environments, in: International Conference on 3D Vision, 3DV, 2017.
[62] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, A. Torralba, Scene parsing through ADE20k dataset, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.

Tatiana Zemskova received an M.S. degree in Applied Mathematics and Computer Science from the Moscow Institute of Physics and Technology, Moscow, Russia, 2023 and an M.S. degree in Engineering from Ecole Polytechnique, Palaiseau, France, 2023. She is currently pursuing a Ph.D. degree in computer science at the Moscow Institute of Physics and Technology, Moscow, Russia. From 2024 to the present, she has been working as a Junior Research Scientist at the Artificial Intelligence Research Institute, Moscow, Russia. Her research interests include computer vision, embodied AI and robotic systems.

Margarita Kichik received an B.S. degree in Applied Mathematics and Physics from the Moscow Institute of Physics and Technology, Moscow, Russia, 2023. She is currently pursuing an M.S. degree in Applied Mathematics and Computer Science from the Moscow Institute of Physics and Technology, Moscow, Russia.

From 2022 to the present, she has been working as an Engineer at the Moscow Institute of Physics and Technology, Moscow, Russia. Her research interests include computer vision and robotic systems.

Dmitry A. Yudin received the engineering diploma in automation of technological processes and production in 2010 and the Ph.D. degree in computer science from the Belgorod State Technological University (BSTU) named after V.G. Shukhov, Belgorod, Russia in 2014. From 2009 to 2019, he was a Researcher and Assistant Professor with Technical Cybernetics Department at BSTU n.a. V.G. Shukhov. Since 2019, he has been the head of the Intelligent Transport Laboratory at the Moscow Institute of Physics and Technology, Moscow, Russia. Since 2021, he has been a Senior Researcher at AIRI (Artificial Intelligence Research Institute), Moscow, Russia.

He is the author of more than 100 articles. His research interests include computer vision, deep learning, and robotics.

Aleksei Staroverov received an M.S. degree from Bauman Moscow State Technical University, Moscow, Russia in 2019. He is currently pursuing a Ph.D. degree in computer science at the Moscow Institute of Physics and Technology, Moscow, Russia. His research thesis involves the methods and algorithms for the automatic determination of subgoals in a reinforcement learning problem for robotic systems.

From 2022 to the present, he has been working as a Researcher at the Artificial Intelligence Research Institute, Moscow, Russia. His research interests include reinforcement learning, deep learning, and robotic systems.

Aleksandr I. Panov earned an M.S. in Computer Science from the Moscow Institute of Physics and Technology, Moscow, Russia, 2011 and a Ph.D. in Theoretical Computer Science from the Institute for Systems Analysis, Moscow, Russia, in 2015. In 2024 he defended the thesis for the degree of Doctor of Science in AI and ML.

Since 2010, he has been a research fellow with the Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, Moscow, Russia. Since 2018, he has headed the Cognitive Dynamic System Laboratory at the Moscow Institute of Physics and Technology, Moscow, Russia. He authored three books and more than 100 research papers. In 2021, he joined the research group on Neurosymbolic Integration at the Artificial Intelligence Research Institute, Moscow, Russia. His academic focus areas include behavior planning, reinforcement learning, embodied AI, and cognitive robotics.