

Semantic Curiosity for Active Visual Learning

Devendra Singh Chaplot^{*1}, Helen Jiang^{*1}, Saurabh Gupta², and Abhinav Gupta¹

¹ Carnegie Mellon University
{chaplot,helenjia,abhinav}@cs.cmu.edu
² UIUC
saurabhg@illinois.edu

Webpage: <https://devdrachaplot.github.io/projects/SemanticCuriosity>

Abstract. In this paper, we study the task of embodied interactive learning for object detection. Given a set of environments (and some labeling budget), our goal is to learn an object detector by having an agent select what data to obtain labels for. How should an exploration policy decide which trajectory should be labeled? One possibility is to use a trained object detector’s failure cases as an external reward. However, this will require labeling millions of frames required for training RL policies, which is infeasible. Instead, we explore a self-supervised approach for training our exploration policy by introducing a notion of semantic curiosity. Our semantic curiosity policy is based on a simple observation – the detection outputs should be consistent. Therefore, our semantic curiosity rewards trajectories with inconsistent labeling behavior and encourages the exploration policy to explore such areas. The exploration policy trained via semantic curiosity generalizes to novel scenes and helps train an object detector that outperforms baselines trained with other possible alternatives such as random exploration, prediction-error curiosity, and coverage-maximizing exploration.

Keywords: Embodied Learning, Active Visual Learning, Semantic Curiosity, Exploration

1 Introduction

Imagine an agent whose goal is to learn how to detect and categorize objects. How should the agent learn this task? In the case of humans (especially babies), learning is quite interactive in nature. We have the knowledge of what we know and what we don’t, and we use that knowledge to guide our future experiences/supervision. Compare this to how current algorithms learn – we create datasets of random images from the internet and label them, followed by model learning. The model has no control over what data and what supervision it gets – it is resigned to the static biased dataset of internet images. Why does current learning look quite different from how humans learn? During the early 2000s, as data-driven approaches started to gain acceptance, the computer vision community struggled with comparisons and knowing which approaches work and which

* Equal Contribution

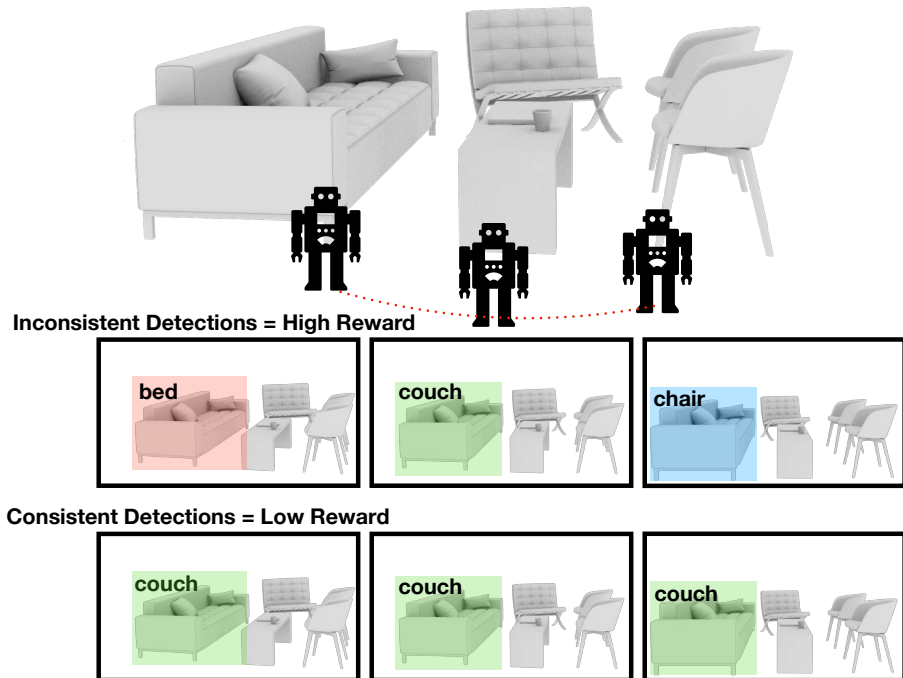


Fig. 1: Semantic Curiosity: We propose semantic curiosity to learn exploration for training object detectors. Our semantically curious policy attempts to take actions such that the object detector will produce inconsistent outputs.

don't. As a consequence, the community introduced several benchmarks from BSDS [34] to VOC [18]. However, a negative side effect of these benchmarks was the use of static training and test datasets. While the pioneering works in computer vision focused on active vision and interactive learning, most of the work in the last two decades focuses on static internet vision. But as things start to work on the model side, we believe it is critical to look at the big picture again and return our focus to an embodied and interactive learning setup.

In an embodied interactive learning setup, an agent has to perform actions such that observations generated from these actions can be useful for learning to perform the semantic task. Several core research questions need to be answered: (a) what is the policy of exploration that generates these observations? (b) what should be labeled in these observations - one object, one frame, or the whole trajectory? (c) and finally, how do we get these labels? In this paper, we focus on the first task - what should the exploration policy be to generate observations which can be useful in training an object detector? Instead of using labels, we focus on learning these trajectories in an unsupervised/self-supervised manner. Once the policy has been learned, we use the policy in novel (previously unseen) scenes to perform actions. As observations are generated, we assume that an oracle will densely label all the objects of interest in the trajectories.

So what are the characteristics of a good exploration policy for visual learning, and how do we learn it? A good semantic exploration policy is one which generates observations of objects and not free-space or the wall/ceiling. But not only should the observations be objects, but a good exploration policy should also observe many unique objects. Finally, a good exploration policy will move to parts of the observation space where the current object detection model fails or does not work. Given these characteristics, how should we define a reward function that could be used to learn this exploration policy? Note, as one of the primary requirements, we assume the policy is learned in a self-supervised manner – that is, we do not have ground-truth objects labeled which can help us figure out where the detections work or fail.

Inspired by recent work in intrinsic motivation and curiosity for training policies without external rewards [37, 38], we propose a new intrinsic reward called semantic curiosity that can be used for the exploration and training of semantic object detectors. In the standard curiosity reward, a policy is rewarded if the predicted future observation does not match the true future observation. The loss is generally formulated in the pixel-based feature space. A corresponding reward function for semantic exploration would be to compare semantic predictions with the current model and then confirm with ground-truth labels – however, this requires external labels (and hence is not self-supervised anymore). Instead, we formulate semantic curiosity based on the meta-supervisory signal of consistency in semantic prediction – that is, if our model truly understands the object, it should predict the same label for the object even as we move around and change viewpoints. Therefore, we exploit consistency in label prediction to reward our policies. Our semantic curiosity rewards trajectories which lead to inconsistent labeling behavior of the same object by the semantic classifier. Our experiments indicate that training an exploration policy via semantic curiosity generalizes to novel scenes and helps train an object detector which outperforms baselines trained with other possible alternatives such as random exploration, pixel curiosity, and free space/map curiosity. We also perform a large set of experiments to understand the behavior of a policy trained with semantic curiosity.

2 Related Work

We study the problem of how to sample training data in embodied contexts. This is related to active learning (picking what sample to label), active perception (how to move around to gain more information), intrinsic motivation (picking what parts of the environment to explore). Learning in embodied contexts can also leverage spatio-temporal consistency. We survey these research areas below.

Active Perception. Active perception [5] refers to the problem of actively moving the sensors around at *test time* to improve performance on the task by gaining more information about the environment. This has been instantiated in the context of object detection [1], amodal object detection [56], scene completion [29], and localization [11, 22]. We consider the problem in a different setting and study how to efficiently move around to best *learn* a model. Furthermore,

our approach to learn this movement policy is self-supervised and does not rely on end-task rewards, which were used in [11, 29, 56].

Active Learning. Our problem is more related to that of active learning [45], where an agent actively acquires labels for unlabeled data to improve its model at the fastest rate [24, 44, 57]. This has been used in a number of applications such as medical image analysis [30], training object detectors [50, 55], video segmentation [21], and visual question answering [36]. Most works tackle the setting in which the unlabeled data has already been collected. In contrast, we study learning a policy for efficiently acquiring effective unlabeled data, which is complementary to such active learning efforts.

Intrinsic Rewards. Our work is also related to work on exploration in reinforcement learning [3, 28, 42, 48]. The goal of these works is to effectively explore a Markov Decision Process to find high reward paths. A number of works formulate this as a problem of maximizing an intrinsic reward function which is designed to incentivize the agent to seek previously unseen [19] or poorly understood [37] parts of the environment. This is similar to our work, as we also seek poorly understood parts of the environment. However, we measure this understanding via multi-view consistency in semantics. This is in a departure from existing works that measure it in 2D image space [37], or consistency among multiple models [38]. Furthermore, our focus is not effective exhaustive exploration, but exploration for the purpose of improving semantic models.

Spatio-Temporal smoothing at test time. A number of papers use spatio-temporal consistency at test time for better and more consistent predictions [7, 23]. Much like the distinction from active perception, our focus is using it to generate better data at train time.

Spatio-temporal consistency as training signal. Labels have been propagated in videos to simplify annotations [51], improve prediction performance given limited data [4, 6], as well as collect images [15]. This line of work leverages spatio-temporal consistency to propagate labels for more efficient labeling. Researchers have also used multi-view consistency to learn about 3D shape from weak supervision [49]. We instead leverage spatio-temporal consistency as a cue to identify what the model does not know. [46] is more directly related, but we tackle the problem in an embodied context and study how to navigate to gather the data, rather than analyzing passive datasets for what labels to acquire.

Visual Navigation and Exploration. Prior work on visual navigation can broadly be categorized into two classes based on whether the location of the goal is known or unknown. Navigation scenarios, where the location of the goal is known, include the most common *pointgoal* task where the coordinate to the goal is given [26, 40]. Another example of a task in this category is vision and language navigation [2] where the path to the goal is described in natural language. Tasks in this category do not require exhaustive exploration of the environment as the location of the goal is known explicitly (coordinates) or implicitly (path).

Navigation scenarios, where the location of the goal is not known, include a wide variety of tasks. These include navigating to a fixed set of objects [10, 17, 26, 31, 35, 53], navigating to an object specified by language [13, 27] or by

an image [12, 58], and navigating to a set of objects in order to answer a question [16, 25]. Tasks in this second category essentially involve efficiently and exhaustively exploring the environment to search the desired object. However, most of the above approaches overlook the exploration problem by spawning the target a few steps away from the goal and instead focus on other challenges. For example, models for playing FPS games [10, 17, 31, 53] show that end-to-end RL policies are effective at reactive navigation and short-term planning such as avoiding obstacles and picking positive reward objects as they randomly appear in the environment. Other works show that learned policies are effective at tackling challenges such as perception (in recognizing the visual goal) [12, 58], grounding (in understanding the goal described by language) [13, 27] or reasoning (about visual properties of the target object) [16, 25]. While end-to-end reinforcement learning is shown to be effective in addressing these challenges, they are ineffective at exhaustive exploration and long-term planning in a large environment as the exploration search space increases exponentially as the distance to the goal increases.

Some very recent works explicitly tackle the problem of exploration by training end-to-end RL policies maximizing the explored area [9, 14, 20]. The difference between these approaches and our method is twofold: first, we train semantically-aware exploration policies as compared spatial coverage maximization in some prior works [9, 14], and second, we train our policy in an unsupervised fashion, without requiring any ground truth labels from the simulator as compared to prior works trained using rewards based on ground-truth labels [20].

3 Overview

Our goal is to learn an exploration policy such that if we use this policy to generate trajectories in a novel scene (and hence observations) and train the object detector from the trajectory data, it would provide a robust, high-performance detector. In literature, most approaches use on-policy exploration; that is, they use the external reward to train the policy itself. However, training an action policy to sample training data for object recognition requires labeling objects. Specifically, these approaches would use the current detector to predict objects and compare them to the ground-truth; they reward the policy if the predictions do not match the ground-truth (the policy is being rewarded to explore regions where the current object detection model fails). However, training such a policy via semantic supervision and external rewards would have a huge bottleneck of supervision. Given that our RL policies require millions of samples (in our case, we train using 10M samples), using ground-truth supervision is clearly not the way. What we need is an intrinsic motivation reward that can help train a policy which can help sample training data for object detectors.

We propose a semantic curiosity formulation. Our work is inspired by a plethora of efforts in active learning [45] and recent work on intrinsic reward using disagreement [38]. The core idea is simple – a good object detector has not only high mAP performance but is also consistent in predictions. That is,

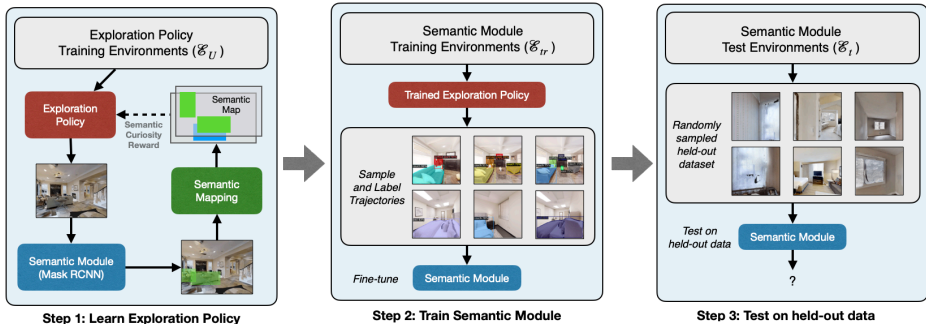


Fig. 2: Embodied Active Visual Learning: We use semantic curiosity to learn an exploration policy on \mathcal{E}_U scenes. The exploration policy is learned by projecting segmentation masks on the top-down view to create semantic maps. The entropy of semantic map defines the inconsistency of the object detection module. The learned exploration policy is then used to generate training data for the object detection/segmentation module. The labeled data is then used to finetune and evaluate the object detection/segmentation.

the detector should predict the same label for different views of the same object. We use this meta-signal of consistency to train our action policy by rewarding trajectories that expose inconsistencies in an object detector. We measure inconsistencies by measuring temporal entropy of prediction – that is, if an object is labeled with different classes as the viewpoint changes, it will have high temporal entropy. The trajectories with high temporal entropy are then labeled via an oracle and used as the data to retrain the object detector (See Figure 2).

4 Methodology

Consider an agent \mathcal{A} which can perform actions in environments \mathcal{E} . The agent has an exploration policy $a_t = \pi(x_t, \theta)$ that predicts the action that the agent must take for current observation x_t . θ represents the parameters of the policy that have to be learned. The agent also has an object detection model \mathcal{O} which takes as input an image (the current observation) and generates a set of bounding boxes along with their categories and confidence scores.

The goal is to learn an exploration policy π , which is used to sample N trajectories $\tau_1 \dots \tau_N$ in a set of novel environments (and get them semantically labeled). When used to train an object detector, this labeled data would yield a high-performance object detector. In our setup, we divide the environments into three non-overlapping sets ($\mathcal{E}_U, \mathcal{E}_{tr}, \mathcal{E}_t$) – the first set is the set of environments where the agent will learn the exploration policy π , the second set is the object detection training environments where we use π to sample trajectories and label them, and the third set is the test environment where we sample random images and test the performance of the object detector on those images.

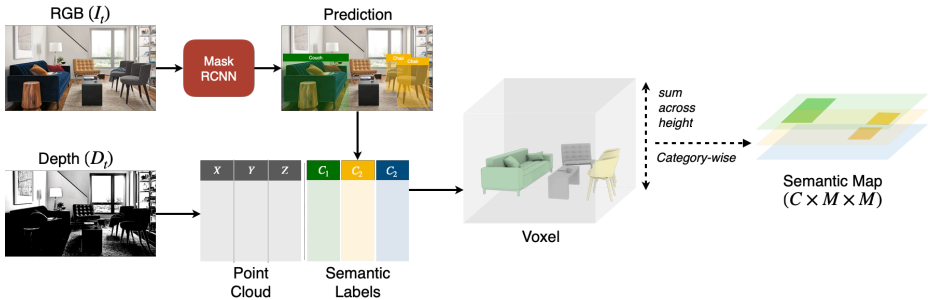


Fig. 3: Semantic Mapping. The Semantic Mapping module takes in a sequence of RGB (I_t) and Depth (D_t) images and produces a top-down Semantic Map.

4.1 Semantic Curiosity Policy

We define semantic curiosity as the temporal inconsistency in object detection and segmentation predictions from the current model. We use a Mask RCNN to obtain the object predictions. In order to associate the predictions across frames in a trajectory, we use a semantic mapping module as described below.

Semantic Mapping. The Semantic Mapping module takes in a sequence of RGB (I_t) and Depth (D_t) images and produces a top-down semantic map (M_t^{Sem}) represented by a 3-dimensional tensor $C \times M \times M$, where M is the length of the square top-down map, and C is the number of semantic categories. Each element (c, i, j) in this semantic map is 1 if the Mask RCNN predicted the object category c at coordinates (i, j) on the map in any frame during the whole trajectory and 0 otherwise. Figure 3 shows how the semantic map is generated for a single frame. The input RGB frame (I_t) is passed through a current Mask RCNN to obtain object segmentation predictions, while the Depth frame is used to calculate the point cloud. Each point in the point cloud is associated with its semantic labels based on Mask RCNN predictions. Note that these are not ground-truth labels, as each pixel is assigned the category of the highest-confidence Mask RCNN segmentation prediction on the corresponding pixel. The point cloud with the associated semantic labels is projected to a 3-dimensional voxel map using geometric computations. The voxel representation is converted to a top-down map by max-pooling the values across the height. The resulting 2D map is converted to a 3-dimensional Semantic Map, such that each channel represents an object category.

The above gives a first-person egocentric projection of the semantic map at each time-step. The egocentric projections at each time step are used to compute a geocentric map over time using a spatial transformation technique similar to Chaplot et al. [9]. The egocentric projections are converted to the geocentric projections by doing a spatial transformation based on the agent pose. The semantic map at each time step is computed by pooling the semantic map at the previous timestep with the current geocentric prediction. Please refer to [9] for more details on these transformations.

Semantic Curiosity Reward. The semantic map allows us to associate the object predictions across different frames as the agent is moving. We define the semantic curiosity reward based on the temporal inconsistency of the object predictions. If an object is predicted to have different labels across different frames, multiple channels in the semantic map at the coordinates corresponding to the object will have 1s. Such inconsistencies are beneficial for visual learning in downstream tasks, and hence, favorable for the semantic curiosity policy. Thus, we define the cumulative semantic curiosity reward to be proportional to the sum of all the elements in the semantic map. Consequently, the semantic curiosity reward per step is just the increase in the sum of all elements in the semantic map as compared to the previous time step:

$$r_{SC} = \lambda_{SC} \sum_{(c,i,j) \in (C,M,M)} (M_t^{Sem}[c, i, j] - M_{t-1}^{Sem}[c, i, j])$$

where λ_{SC} is the semantic curiosity reward coefficient. Summation over the channels encourages exploring frames with temporally inconsistent predictions. Summation across the coordinates encourages exploring as many objects with temporally inconsistent predictions as possible.

The proposed Semantic Curiosity Policy is trained using reinforcement learning to maximize the cumulative semantic curiosity reward. Note that although the depth image and agent pose are used to compute the semantic reward, we train the policy only on RGB images.

5 Experimental Setup

We use the Habitat simulator [41] with three different datasets for our experiments: Gibson [54], Matterport [8] and Replica [47]. While the RGB images used in our experiments are visually realistic as they are based on real-world reconstructions, we note that the agent pose and depth images in the simulator are noise-free unlike the real-world. Prior work has shown that both depth and agent pose can be estimated effectively from RGB images under noisy odometry [9]. In this paper, we assume access to perfect agent pose and depth images, as these challenges are orthogonal to the focus of this paper. Furthermore, these assumptions are only required in the unsupervised pre-training phase for calculating the semantic curiosity reward and not at inference time when our trained semantic-curiosity policy (based only on RGB images) is used to gather exploration trajectories for training the object detector.

In a perfectly interactive learning setup, the current model’s uncertainty will be used to sample a trajectory in a new scene, followed by labeling and updating the learned visual model (Mask-RCNN). However, due to the complexity of this online training mechanism, we show results on batch training. We use a pre-trained COCO Mask-RCNN as an initial model and train the exploration policy on that model. Once the exploration policy is trained, we collect trajectories in the training environments and then obtain the labels on these trajectories. The labeled examples are then used to fine-tune the Mask-RCNN detector.

5.1 Implementation details

Exploration Policy: We train our semantic curiosity policy on the Gibson dataset and test it on the Matterport and Replica datasets. We train the policy on the set of 72 training scenes in the Gibson dataset specified by Savva et al. [41]. Our policy is trained with reinforcement learning using Proximal Policy Optimization [43]. The policy architecture consists of convolutional layers of a pre-trained ResNet18 visual encoder, followed by two fully connected layers and a GRU layer leading to action distribution as well as value prediction. We use 72 parallel threads (one for each scene) with a time horizon on 100 steps and 36 mini batches per PPO epoch. The curiosity reward coefficient is set to $\lambda_{SC} = 2.5 \times 10^{-3}$. We use an entropy coefficient of 0.001, the value loss coefficient of 0.5. We use Adam optimizer with a learning rate of 1×10^{-5} . The maximum episode length during training is 500 steps.

Fine-tuned Object Detector: We consider 5 classes of objects, chosen because they overlap with the COCO dataset [33] and correspond to objects commonly seen in an indoor scene: chair, bed, toilet, couch, and potted plant. To start, we pre-train a Faster-RCNN model [39] with FPN [32] using ResNet-50 as the backbone on the COCO dataset labeled with these 5 overlapping categories. We then fine-tuned our models on the trajectories collected by the exploration policies for 90000 iterations using a batch size of 12 and a learning rate of 0.001, with annealing by a factor of 0.1 at iterations 60000 and 80000. We use the Detectron2 codebase [52] and set all other hyperparameters to their defaults in this codebase. We compute the AP50 score (i.e., average precision using an IoU threshold of 50) on the validation set every 5000 iterations.

5.2 Baselines

We use a range of baselines to gather exploration trajectories and compare them to the proposed Semantic Curiosity policy:

- **Random.** A baseline sampling actions randomly.
- **Prediction Error Curiosity.** This baseline is based on Pathak et al. [37], which trains an RL policy to maximize error in a forward prediction model.
- **Object Exploration.** Object Exploration is a naive baseline where an RL policy is trained to maximize the number of pre-trained Mask R-CNN detections. The limitation of simply maximizing the number of detections is that the policy can learn to look at frames with more objects but might not learn to look at different objects across frames or objects with low confidence.
- **Coverage Exploration.** This baseline is based on Chen et al. [14], where an RL policy is trained to maximize the total explored area.
- **Active Neural SLAM.** This baseline is based on Chaplot et al. [9] and uses a modular and hierarchical system to maximize the total explored area.

After training the proposed policy and the baselines in the Gibson domain, we use them directly (without fine-tuning) in the Matterport and Replica domains. We sample trajectories using each exploration policy, using the images and ground-truth labels to train an object detection model.



Fig. 4: Temporal Inconsistency Examples. Figure showing example trajectories sampled from the semantic curiosity exploration policy. We highlight the segmentation/detection inconsistencies of Mask RCNN. By obtaining labels for these images, the Mask RCNN pipeline improves the detection performance significantly.

6 Analyzing Learned Exploration Behavior

Before we measure the quality of the learned exploration policy for the task of detection/segmentation, we first want to analyze the behavior of the learned policy. This will help characterize the quality of data that is gathered by the exploration policy. We will compare the learned exploration policy against the baselines described above. For all the experiments below, we trained our policy on Gibson scenes and collected statistics in 11 Replica scenes.

Figure 4 shows three sampled trajectories. The pre-trained Mask-RCNN detections are also shown in the observation images. Semantic curiosity prefers trajectories with inconsistent detections. For example, in the top row, the chair and couch detector fire on the same object. In the middle row, the chair is misclassified as a toilet and there is inconsistent labeling in the last trajectory. The bed is misclassified as a couch. By selecting these trajectories and obtaining their labels from an oracle, our approach learns to improve the object detection module.

Table 1: Analysis. Comparing the proposed Semantic Curiosity policy with the baselines along different exploration metrics.

| Method Name | Semantic Curiosity Reward | Explored Area | Number of Object Detections |
|---------------------------|------------------------------|------------------|--------------------------------|
| Random | 1.631 | 4.794 | 82.83 |
| Curiosity [37] | 2.891 | 6.781 | 112.24 |
| Object exploration reward | 2.168 | 6.082 | 382.27 |
| Coverage Exploration [14] | 3.287 | 10.025 | 203.73 |
| Active Neural SLAM [9] | 3.589 | 11.527 | 231.86 |
| Semantic Curiosity | 4.378 | 9.726 | 291.78 |

Table 2: Quality of object detection on training trajectories. We also analyze the training trajectories in terms of how well the pre-trained object detection model works on the trajectories. We want the exploration policy to sample hard data where the pre-trained object detector fails. Data on which the pre-trained model already works well would not be useful for fine-tuning. Thus, lower performance is better.

| Method Name | Chair | Bed | Toilet | Couch | Potted Plant | Average |
|---------------------------|-------|------|--------|-------|--------------|---------|
| Random | 46.7 | 28.2 | 46.9 | 60.3 | 39.1 | 44.24 |
| Curiosity [37] | 49.4 | 18.3 | 1.8 | 67.7 | 49.0 | 37.42 |
| Object Exploration | 54.3 | 24.8 | 5.7 | 76.6 | 49.6 | 42.2 |
| Coverage Exploration [14] | 48.5 | 23.1 | 69.2 | 66.3 | 48.0 | 51.02 |
| Active Neural SLAM [9] | 51.3 | 20.5 | 49.4 | 59.7 | 45.6 | 45.3 |
| Semantic Curiosity | 51.6 | 14.6 | 14.2 | 65.2 | 50.4 | 39.2 |

Table 1 shows the behavior of all of the policies on three different metrics. The first metric is the semantic curiosity reward itself which measures uncertain detections in the trajectory data. Since our policy is trained for this reward, it gets the highest score on the sampled trajectories. The second metric is the amount of explored area. Both [14] and [9] optimize this metric and hence perform the best (they cover a lot of area but most of these areas will either not have objects or not enough contradictory overlapping detections). The third metric is the number of objects in the trajectories. The object exploration baseline optimizes for this reward and hence performs the best but it does so without exploring diverse areas or uncertain detections/segmentations. If we look at the three metrics together it is clear that our policy has the right tradeoff – it explores a lot of area but still focuses on areas where objects can be detected. Not only does it find a large number of object detections, but our policy also prefers inconsistent object detections and segmentations. In Figure 5, we show some examples of trajectories seen by the semantic curiosity exploration along with the semantic map. It shows examples of the same object having different object predictions from different viewpoints and also the representation in the semantic map. In Figure 6, we show a qualitative comparison of maps and objects explored by the proposed model and all the baselines. Example trajectories

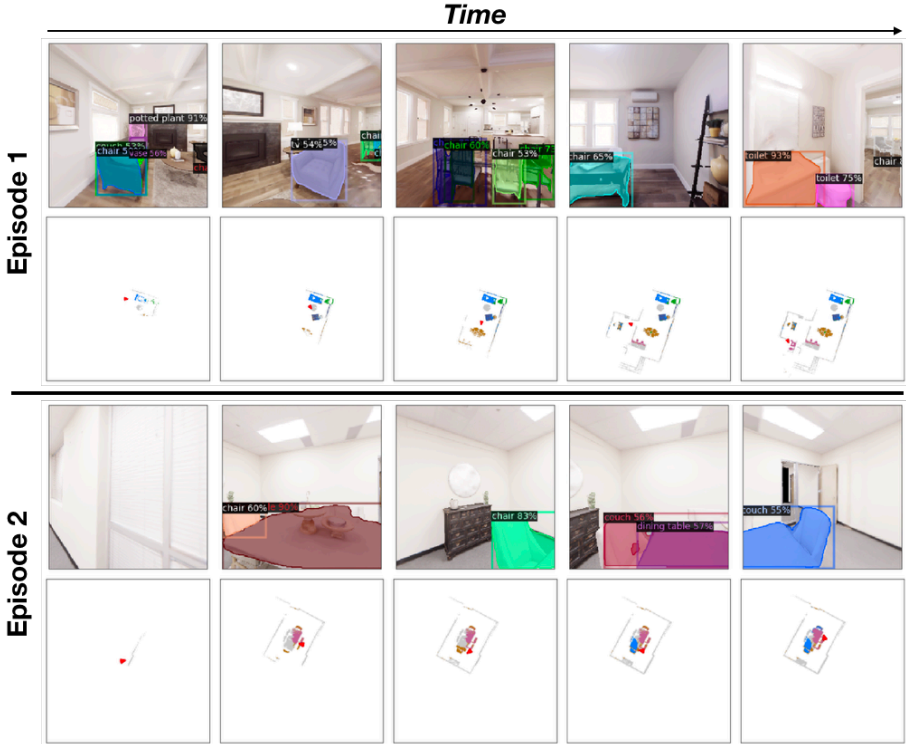


Fig. 5: Example trajectories. Figure showing example trajectories sampled from the semantic curiosity exploration policy. In each episode the top row shows the first-person images seen by the agent and the pre-trained Mask R-CNN predictions. The bottom rows show a visualization of the semantic map where colors denote different object categories. Different colors for the same object indicate that the same object is predicted to have different categories from different view points.

in this figure indicate that the semantic curiosity policy explores more unique objects with higher temporal inconsistencies.

Next, we analyze the trajectories created by different exploration policies during the object detection training stage. Specifically, we want to analyze the kind of data that is sampled by these trajectories. How is the performance of a pre-trained detector on this data? If the pre-trained detector already works well on the sampled trajectories, we would not see much improvement in performance by fine-tuning with this data. In Table 2, we show the results of this analysis for these trajectories. As the results indicate, the mAP50 score is low for the data obtained by the semantic curiosity policy.³ As the pre-trained object detector fails more on the data sampled by semantic curiosity, labeling this data would intuitively improve the detection performance.

³ Note that curiosity-based policy has the lowest mAP because of outlier toilet category.

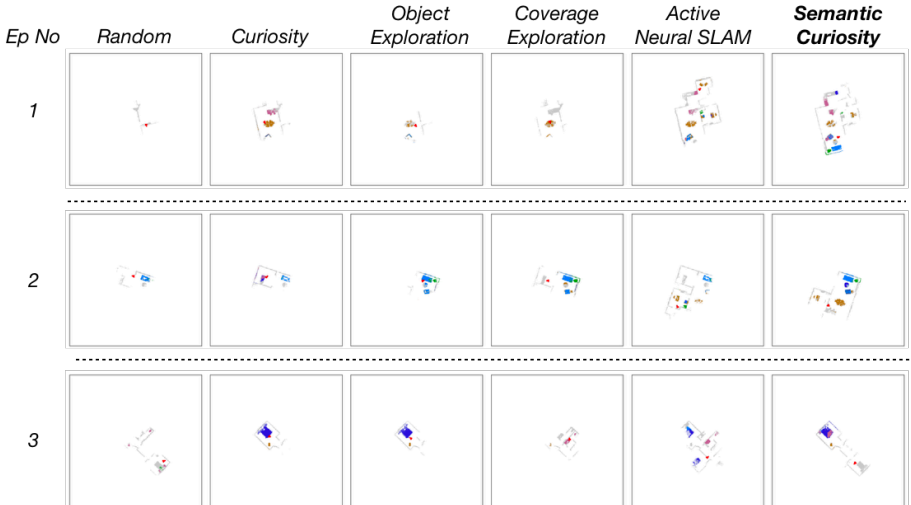


Fig. 6: Qualitative Comparison. Figure showing map and objects explored by the proposed Semantic Curiosity policy and the baselines in 3 example episodes. Semantic Curiosity Policy explores more unique objects with higher temporal inconsistency (denoted by different colors for the same object).

7 Actively Learned Object Detection

Finally, we evaluate the performance of our semantic curiosity policy for the task of object detection. The semantic curiosity exploration policy is trained on 72 Gibson scenes. The exploration policy is then used to sample data on 50 Matterport scenes. Finally, the learned object detector is tested on 11 Matterport scenes. For each training scene, we sample 5 trajectories of 300 timesteps leading to 75,000 total training images with ground-truth labels. For test scenes, we randomly sample images from test scenes.

In Table 3, we report the top AP50 scores for each method. Our results demonstrate that the proposed semantic curiosity policy obtains higher quality data for performing object detection tasks over alternative methods of exploration. First, we outperform the policy that tries to see maximum coverage area (and hence the most novel images). Second, our approach also outperforms the policy that detects a lot of objects. Finally, apart from outperforming the random policy, visual curiosity [37], and coverage; we also outperform the highly-tuned approach of [9]. The underlying algorithm is tuned on this data and was the winner of the RGB and RGBD challenge in Habitat.

8 Conclusion and Future Work

We argue that we should go from detection/segmentation driven by static datasets to a more embodied active learning setting. In this setting, an agent can move in the scene and create its own datapoints. An oracle labels these datapoints and helps the agent learn a better semantic object detector. This setting is closer to

Table 3: Object Detection Results. Object detection results in the Matterport domain using the proposed Semantic Curiosity policy and the baselines. We report AP50 scores on randomly sampled images in the test scenes. Training on data gathered from the semantic curiosity trajectories results in improved object detection scores.

| Method Name | Chair | Bed | Toilet | Couch | Potted Plant | Average |
|---------------------------|-------|------|--------|-------|--------------|--------------|
| PreTrained | 41.8 | 17.3 | 34.9 | 41.6 | 23.0 | 31.72 |
| Random | 51.7 | 17.2 | 43.0 | 45.1 | 30.0 | 37.4 |
| Curiosity [37] | 48.4 | 18.5 | 42.3 | 44.3 | 32.8 | 37.26 |
| Object Exploration | 50.3 | 16.4 | 40.0 | 39.7 | 29.9 | 35.26 |
| Coverage Exploration [14] | 50.0 | 19.1 | 38.1 | 42.1 | 33.5 | 36.56 |
| Active Neural SLAM [9] | 53.1 | 19.5 | 42.0 | 44.5 | 33.4 | 38.5 |
| Semantic Curiosity | 52.3 | 22.6 | 42.9 | 45.7 | 36.3 | 39.96 |

how humans learn to detect and recognize objects. In this paper, we focus on the exploration policy for sampling images to be labeled. We ask a basic question – how should an agent explore to learn how to detect objects? Should the agent try to cover as many scenes as possible in the hopes of seeing more diverse examples, or should the agent focus on observing as many objects as possible?

We propose semantic curiosity as a reward to train the exploration policy. Semantic curiosity encourages trajectories which will lead to inconsistent detection behavior from an object detector. Our experiments indicate that exploration driven by semantic curiosity shows all of the good characteristics of an exploration policy: uncertain/high entropy detections, attention to objects rather than the entire scene and also high coverage for diverse training data. We also show that an object detector trained on trajectories from a semantic curiosity policy leads to the best performance compared to a plethora of baselines. For future work, this paper is just the first step in embodied active visual learning. It assumes perfect odometry, localization and zero trajectory labeling costs. It also assumes that the trajectories will be labeled – a topic of interest would be to sample trajectories with which minimal labels can learn the best detector. Finally, the current approach is demonstrated in simulators - it will be interesting to see whether the performance can transfer to real-world robots.

Acknowledgements

This work was supported by IARPA DIVA D17PC00340, ONR MURI, ONR Grant N000141812861, ONR Young Investigator, DARPA MCS, and NSF Graduate Research Fellowship. We would also like to thank NVIDIA for GPU support.

Licenses for referenced datasets:

Gibson: http://svl.stanford.edu/gibson2/assets/GDS_agreement.pdf

Matterport3D: http://kaldir.vc.in.tum.de/matterport/MP_TOS.pdf

Replica: <https://github.com/facebookresearch/Replica-Dataset/blob/master/LICENSE>

References

1. Ammirato, P., Poiron, P., Park, E., Košecká, J., Berg, A.C.: A dataset for developing and benchmarking active vision. In: 2017 IEEE International Conference on Robotics and Automation (ICRA). pp. 1378–1385. IEEE (2017)
2. Anderson, P., Wu, Q., Teney, D., Bruce, J., Johnson, M., Sünderhauf, N., Reid, I., Gould, S., van den Hengel, A.: Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3674–3683 (2018)
3. Auer, P.: Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research* **3**(Nov), 397–422 (2002)
4. Badrinarayanan, V., Galasso, F., Cipolla, R.: Label propagation in video sequences. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 3265–3272. IEEE (2010)
5. Bajcsy, R.: Active perception. *Proceedings of the IEEE* **76**(8), 966–1005 (1988)
6. Bengio, Y., Delalleau, O., Le Roux, N.: Label propagation and quadratic criterion. In: *Semi-Supervised Learning* (2006)
7. Chandra, S., Couprie, C., Kokkinos, I.: Deep spatio-temporal random fields for efficient video segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8915–8924 (2018)
8. Chang, A., Dai, A., Funkhouser, T., Halber, M., Niessner, M., Savva, M., Song, S., Zeng, A., Zhang, Y.: Matterport3d: Learning from rgb-d data in indoor environments. *International Conference on 3D Vision (3DV)* (2017)
9. Chaplot, D.S., Gandhi, D., Gupta, S., Gupta, A., Salakhutdinov, R.: Learning to explore using active neural slam. In: ICLR (2020), <https://openreview.net/forum?id=HklXn1BKDH>
10. Chaplot, D.S., Lample, G.: Arnold: An autonomous agent to play fps games. In: *Thirty-First AAAI Conference on Artificial Intelligence* (2017)
11. Chaplot, D.S., Parisotto, E., Salakhutdinov, R.: Active neural localization. In: *International Conference on Learning Representations* (2018), https://openreview.net/forum?id=ry6-G_66b
12. Chaplot, D.S., Salakhutdinov, R., Gupta, A., Gupta, S.: Neural topological slam for visual navigation. In: *CVPR* (2020)
13. Chaplot, D.S., Sathyendra, K.M., Pasumarthi, R.K., Rajagopal, D., Salakhutdinov, R.: Gated-attention architectures for task-oriented language grounding. In: *Thirty-Second AAAI Conference on Artificial Intelligence* (2018)
14. Chen, T., Gupta, S., Gupta, A.: Learning exploration policies for navigation. In: *International Conference on Learning Representations* (2019), <https://openreview.net/forum?id=SyMwn05F7>
15. Chen, X., Shrivastava, A., Gupta, A.: Neil: Extracting visual knowledge from web data. In: *Proceedings of the IEEE International Conference on Computer Vision*. pp. 1409–1416 (2013)
16. Das, A., Datta, S., Gkioxari, G., Lee, S., Parikh, D., Batra, D.: Embodied question answering. In: *CVPR* (2018)
17. Dosovitskiy, A., Koltun, V.: Learning to act by predicting the future. In: *ICLR* (2017)
18. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. <http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html>

19. Eysenbach, B., Gupta, A., Ibarz, J., Levine, S.: Diversity is all you need: Learning skills without a reward function. In: International Conference on Learning Representations (2019), <https://openreview.net/forum?id=SJx63jRqFm>
20. Fang, K., Toshev, A., Fei-Fei, L., Savarese, S.: Scene memory transformer for embodied agents in long-horizon tasks. In: CVPR (2019)
21. Fathi, A., Balcan, M.F., Ren, X., Rehg, J.M.: Combining self training and active learning for video segmentation. In: Proceedings of the British Machine Vision Conference. Georgia Institute of Technology (2011)
22. Fox, D., Burgard, W., Thrun, S.: Active markov localization for mobile robots. *Robotics and Autonomous Systems* **25**(3-4), 195–207 (1998)
23. Gadde, R., Jampani, V., Gehler, P.V.: Semantic video cnns through representation warping. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4453–4462 (2017)
24. Gal, Y., Islam, R., Ghahramani, Z.: Deep bayesian active learning with image data. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 1183–1192. JMLR. org (2017)
25. Gordon, D., Kembhavi, A., Rastegari, M., Redmon, J., Fox, D., Farhadi, A.: Iqa: Visual question answering in interactive environments. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4089–4098 (2018)
26. Gupta, S., Davidson, J., Levine, S., Sukthankar, R., Malik, J.: Cognitive mapping and planning for visual navigation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2616–2625 (2017)
27. Hermann, K.M., Hill, F., Green, S., Wang, F., Faulkner, R., Soyer, H., Szepesvari, D., Czarnecki, W., Jaderberg, M., Teplyashin, D., et al.: Grounded language learning in a simulated 3d world. arXiv preprint arXiv:1706.06551 (2017)
28. Jaksch, T., Ortner, R., Auer, P.: Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research* **11**(Apr), 1563–1600 (2010)
29. Jayaraman, D., Grauman, K.: Learning to look around: Intelligently exploring unseen environments for unknown tasks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1238–1247 (2018)
30. Kuo, W., Häne, C., Yuh, E., Mukherjee, P., Malik, J.: Cost-sensitive active learning for intracranial hemorrhage detection. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 715–723. Springer (2018)
31. Lample, G., Chaplot, D.S.: Playing FPS games with deep reinforcement learning. In: Thirty-First AAAI Conference on Artificial Intelligence (2017)
32. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)
33. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014)
34. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proc. 8th Int’l Conf. Computer Vision. vol. 2, pp. 416–423 (July 2001)
35. Mirowski, P., Pascanu, R., Viola, F., Soyer, H., Ballard, A.J., Banino, A., Denil, M., Goroshin, R., Sifre, L., Kavukcuoglu, K., et al.: Learning to navigate in complex environments. ICLR (2017)
36. Misra, I., Girshick, R., Fergus, R., Hebert, M., Gupta, A., van der Maaten, L.: Learning by Asking Questions. In: CVPR (2018)

37. Pathak, D., Agrawal, P., Efros, A.A., Darrell, T.: Curiosity-driven exploration by self-supervised prediction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 16–17 (2017)
38. Pathak, D., Gandhi, D., Gupta, A.: Self-supervised exploration via disagreement. In: ICML (2019)
39. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. pp. 91–99 (2015)
40. Savva, M., Chang, A.X., Dosovitskiy, A., Funkhouser, T., Koltun, V.: MI-NOS: Multimodal indoor simulator for navigation in complex environments. arXiv:1712.03931 (2017)
41. Savva, M., Kadian, A., Maksymets, O., Zhao, Y., Wijmans, E., Jain, B., Straub, J., Liu, J., Koltun, V., Malik, J., et al.: Habitat: A platform for embodied ai research. In: ICCV (2019)
42. Schmidhuber, J.: A possibility for implementing curiosity and boredom in model-building neural controllers. In: Proc. of the international conference on simulation of adaptive behavior: From animals to animats. pp. 222–227 (1991)
43. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347 (2017)
44. Sener, O., Savarese, S.: Active learning for convolutional neural networks: A core-set approach. In: International Conference on Learning Representations (2018), <https://openreview.net/forum?id=H1aIuk-RW>
45. Settles, B.: Active learning literature survey. Tech. rep., University of Wisconsin-Madison Department of Computer Sciences (2009)
46. Siddiqui, Y., Valentin, J., Nießner, M.: Viewal: Active learning with viewpoint entropy for semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9433–9443 (2020)
47. Straub, J., Whelan, T., Ma, L., Chen, Y., Wijmans, E., Green, S., Engel, J.J., Mur-Artal, R., Ren, C., Verma, S., Clarkson, A., Yan, M., Budge, B., Yan, Y., Pan, X., Yon, J., Zou, Y., Leon, K., Carter, N., Briales, J., Gillingham, T., Mueggler, E., Pesqueira, L., Savva, M., Batra, D., Strasdat, H.M., Nardi, R.D., Goele, M., Lovegrove, S., Newcombe, R.: The Replica dataset: A digital replica of indoor spaces. arXiv preprint arXiv:1906.05797 (2019)
48. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press (1998), <http://www.cs.ualberta.ca/~sutton/book/the-book.html>
49. Tulsiani, S., Zhou, T., Efros, A.A., Malik, J.: Multi-view supervision for single-view reconstruction via differentiable ray consistency. TPAMI (2019)
50. Vijayanarasimhan, S., Grauman, K.: Large-scale live active learning: Training object detectors with crawled data and crowds. International Journal of Computer Vision **108**(1-2), 97–114 (2014)
51. Vondrick, C., Patterson, D., Ramanan, D.: Efficiently scaling up crowdsourced video annotation. International Journal of Computer Vision pp. 1–21 (2013), <http://dx.doi.org/10.1007/s11263-012-0564-1>, 10.1007/s11263-012-0564-1
52. Wu, Y., Kirillov, A., Massa, F., Lo, W.Y., Girshick, R.: Detectron2. <https://github.com/facebookresearch/detectron2> (2019)
53. Wu, Y., Tian, Y.: Training agent for first-person shooter game with actor-critic curriculum learning. In: ICLR (2017)
54. Xia, F., R. Zamir, A., He, Z.Y., Sax, A., Malik, J., Savarese, S.: Gibson Env: real-world perception for embodied agents. In: Computer Vision and Pattern Recognition (CVPR), 2018 IEEE Conference on. IEEE (2018)

55. Yang, J., Lu, J., Lee, S., Batra, D., Parikh, D.: Visual curiosity: Learning to ask questions to learn visual recognition. In: Conference on Robot Learning. pp. 63–80 (2018)
56. Yang, J., Ren, Z., Xu, M., Chen, X., Crandall, D.J., Parikh, D., Batra, D.: Embodied amodal recognition: Learning to move to perceive objects. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2040–2050 (2019)
57. Yoo, D., Kweon, I.S.: Learning loss for active learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 93–102 (2019)
58. Zhu, Y., Mottaghi, R., Kolve, E., Lim, J.J., Gupta, A., Fei-Fei, L., Farhadi, A.: Target-driven visual navigation in indoor scenes using deep reinforcement learning. In: Robotics and Automation (ICRA), 2017 IEEE International Conference on. pp. 3357–3364. IEEE (2017)