ALAN: Autonomously Exploring Robotic Agents in the Real World

Russell Mendonca, Shikhar Bahl, Deepak Pathak
Carnegie Mellon University

***Abstract*— Robotic agents that operate autonomously in the real world need to continuously explore their environment and learn from the data collected, with minimal human supervision. While it is possible to build agents that can learn in such a manner without supervision, current methods struggle to scale to the real world. Thus, we propose ALAN, an autonomously exploring robotic agent, that can perform many tasks in the real world with little training and interaction time. This is enabled by measuring environment change, which reflects object movement and ignores changes in the robot position. We use this metric directly as an environment-centric signal, and also maximize the uncertainty of predicted environment change, which provides agent-centric exploration signal. We evaluate our approach on two different real-world play kitchen settings, enabling a robot to efficiently explore and discover manipulation skills, and perform tasks specified via goal images. Videos can be found at `https://robo-explorer.github.io/`**

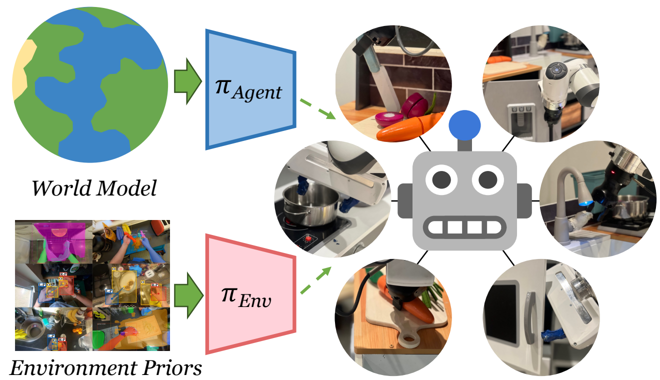


Fig. 1: We present ALAN, an approach for real world robotic exploration in challenging manipulation environments.

I. Introduction

Autonomous robots will need to perform a diverse range of tasks in the real world. Due to the challenges of dealing with uncertainty, deep learning has emerged as a promising approach [1], [2] for robotics. A critical challenge for scaling learning based approaches to more complex settings is the task specification problem. Prior works require heavy reward engineering or human demonstrations, which is not feasible for performing large numbers of tasks. This also requires knowledge of the environment, which might be hard to obtain for every domain. Instead, if robots can collect their own data using task-agnostic objectives, they could then autonomously explore their environments and learn interesting skills.

In the absence of explicit task definitions, the agent should have an efficient way to use all its collected experience for learning. World models [3], [4] provide a means of learning an effective low dimensional representation of raw image observations. Furthermore, if there are certain states where prediction for the world model is difficult, then it likely needs more data for the corresponding part of the environment. This gives rise to a natural intrinsic objective of maximizing model uncertainty [5], [6] for exploration. While this does lead to the discovery of interesting behavior, there has been difficulty in scaling such approaches to real world settings since collecting samples on real hardware is very time-intensive. We ask if there is a different task-agnostic objective that can enable robots to *more efficiently* explore?

In order to address the above question, we present `ALAN`, an efficient autonomous real robotic explorer. Our key insight is that interesting behavior for robots in the manipulation setting mostly involve interactions with objects, which cause changes in the visual features of the observations. Thus, seeking to maximize the change in these visual features can be a useful objective for robots to optimize. Furthermore, if agents learned to model the change in the environment, they can take actions to maximize uncertainty in the *object space* of the environment, as opposed to the full space consisting of both the robot body and the surrounding environment. Seeking to maximize information related to objects in the environment will lead to much more efficient exploration, since the robot will prioritize actions that lead to richer contact interactions. We note that maximizing model uncertainty, (whether in the object space or full image space) is 'agent-centric', since it is dependent on the agent's belief, as opposed to simply maximizing the environment change which is 'environment centric'. The latter is a constant signal agnostic of the agent's internal mental model. We show that leveraging both these objectives can enable a real robot to effectively explore multiple challenging real-world environments, and then perform tasks of interest.

The main contribution of this work is `ALAN`, an efficient real world exploration algorithm, that seeks to take actions that maximize change in the environment, and maximize uncertainty about its internal model of how changes occur in the environment. This approach encourages the robot to interact with objects, and hence collect data relevant to learning manipulation skills faster. We show that our approach enables a real franka robot to effectively explore without any supervision signal in two different, challenging play-kitchen environments using less than 150 interaction trajectories. The robot can then use this data to perform user-specified tasks via goal images in a zero-shot manner, including picking up a knife, and opening a cabinet, fridge or shelf.

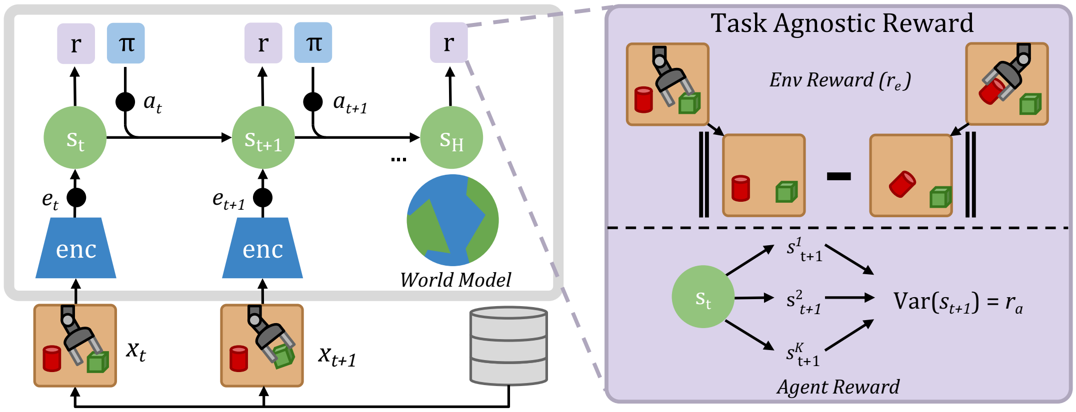


Fig. 2: We propose Autonomous Learning Agents (ALAN) that can enable robots to collect rich data from their environment efficiently. The agent utilizes environment change, both directly as an environment-centric signal, as well as modelling the change and taking actions that maximize uncertainty in change space, which provides agent-centric signal.

II. Related Work

a) Exploration

In reinforcement learning (RL), exploration has been studied in various contexts ranging from tabular settings to high-dimensional continuous spaces. For simple discrete settings, analysis of exploration has included state visitation counts [7] and probability distributions over visited states [8], [9]. For high-dimensional input spaces such as images, previous works have used neural networks to approximate state counts [10], [11], [12]. A common way to describe intrinsic reward for exploration is to use either the error [13] or uncertainty [14], [15] in prediction about how the environment and agent would interact. [16] proposes a differentiable intrinsic reward which measures disagreement using the variance of the prediction of an ensemble of models. [5] leverages a similar disagreement-based intrinsic reward, but explores in the imagination space of a world model [17], [4].

b) Autonomous Learning in the Real World

Training agents in the real world is challenging for a host of reasons, and one of these is the difficulty of providing supervision to the agent. Some prior approaches have designed task specific rewards [18], [1]. However, it is infeasible to define all of the tasks that are possible for the robot to perform, and further there is no guarantee that the designed rewards will allow for the task to be solved efficiently and robustly. There are a number of approaches that provide self-supervision for agents based on mutual information objectives [19], [20], [21], which enables the learning of skill-spaces. However, many of these learned skills are not semantically different and have been difficult to apply to real-world manipulation. Other approaches involve selecting goals from experience. This can directly come from previously seen states [22], from a generative model [23], [24], [25], or from the imagination space of a world-model [6]. While these approaches have shown better results for real-world manipulation, they are still limited in scope, since they require lots of samples for learning. A key reason is that it is difficult for the robot to know *what* to focus on while exploring. Efforts have been made to initialize such approaches from priors of human behavior, such as from internet data [26], [27], [28], however, such methods are not able to learn in an autonomous fashion. Our approach provides an effective new metric that enables efficient self-supervision, and also leverages visual priors to focus on parts of the scene that are more interesting for exploration and discovery of useful skills.

III. Background

a) Model-Based RL and Planning

A Markov Decision Process (MDP) is defined by a set of states $\mathcal{S}$, actions $\mathcal{A}$, transition probabilities between states conditioned on actions, $\mathcal{T}(s_{t+1}|s_t, a_t)$, a initial state distribution $\mathcal{S}_0$, a reward function $\mathcal{R}(s_t, a_t)$. The goal of a model based RL algorithm is to learn a function $f_\phi(s_{t+1}|s_t, a_t)$ which best approximates the the true transition dynamics $\mathcal{T}$ of the MDP. While planning, the Cross-Entropy Method (CEM) is used to find the best set of actions $a_{1:T}$, which produce the highest reward under the trained dynamics model f_ϕ.

b) Sample Efficient Model-Based RL

For efficient model-driven learning, it is desirable to build a general *dynamics model* of the world. When dealing with high-dimensional inputs such as images, it is often beneficial to train the dynamics of the system in a lower dimensional space, to avoid overfitting to visual artifacts. Thus, we employ the Recurrent State-Space Model (RSSM) from [4]. Given an image input x_t at timestep t, we obtain a low dimensional embedding $e_t = E_\phi(x_t)$, where E is the ResNet-based [29] encoder from [30]. RSSMs model the state of the system as s_t (which contains both e_t and a hidden recurrent state h_t) using a forward dynamics model, predicting $s_t = f_\phi(s_{t-1}, a_{t-1})$.

This system is learned via variational inference, leveraging the posterior $q_\phi(s_t|s_{t-1}, a_{t-1}, e_t)$ and an image posterior $p_\phi(x_t|s_t)$, and is trained using the ELBO loss [31], [32].

c) Intrinsic Motivation

When learning a dynamics model of the world, $f(s_{t+1}|s_t, a_t)$, it is possible to use the quality of the model as an intrinsic reward. For instance, [13] uses model prediction error as reward

$$r_t = ||f(s_{t+1}|s_t, a_t) - s_{t+1}||$$

However, this formulation is dependent on environment dynamics, and thus needs a policy-gradient approach to

Fig. 3: Visualizations of the object detections, using [33]. The masks selected to study exploration are the knife, pan and rightcabinet handle from kitchen1 (left), and the topshelf, fridge handles and pot from kitchen2 (right).

optimize it, since future states need to be observed before this metric can be computed. Instead, [16] proposes to minimize the *disagreement* between an ensemble of dynamics model $f_{\phi^{(k)}}$ for $k = 1, ..., M$, which is a fully differentiable objective in terms of the current state and action, which we utilize in our work. The disagreement reward can be described as:

$$\mathbb{E}_{s_t, a_t, s_{t+1} \sim \rho(s)}[\mathrm{Var}_k(f_{\phi^{(k)}})]$$

IV. Autonomous Real World Robot Learning

Intelligent agents should be able to perform diverse tasks in complex, real world environments. There are three major challenges to this: (1) There is a large space of possible interactions, especially in continuous control. (2) It is difficult to obtain any reward signal without human supervision. (3) There is a large cost for collecting data with real hardware.

To this end, we propose `ALAN`, an autonomous robot learning algorithm that is able to efficiently explore in the real world, and learn useful manipulation skills for various objects. `ALAN` defines a novel intrinsic exploration objective for the agent to direct its behavior. This novel objective has an environment-centric component and an agent-centric component. Moreover, we use offline visual data to reduce the search space for the robot, by identifying the locations of potential interesting and complex interactions for the robot.

A. World Model

The robot observations consist of a stream of high-dimensional raw RGB images. These can be effectively processed using world models [3], [34], [35], which learn compact low-dimensional latent spaces that contain temporal information and enable efficient forward prediction. We use the Recurrent state-space model, from [17], [4], [36], which learns latent features with deterministic and stochastic components to model long-range dependencies and uncertainty in the environment respectively. Specifically, the world model has the following networks:

$$\begin{aligned}
&\text{Image Encoder} && h_t = \phi_i(x_t) \\
&\text{Dynamics Prior} && p_\theta(s_{t+1}|s_t, a_t) \\
&\text{Image Decoder} && f_\theta(x_t|s_t) \\
&\text{Dynamics Posterior} && q_\theta(s_{t+1}|s_t, a_t, h_{t+1}) \\
&\text{Embed Decoder} && g_\theta(e_t|s_t)
\end{aligned} \tag{1}$$

The latent features are trained to reconstruct image observations, while also preserving dynamics information. In addition to providing useful representations for control, world models also provides a means for agents to drive their own behavior in the absence of supervision. This involves taking actions that maximize the uncertainty of model predictions [16], [5], [6], leading to information gain for the agent. Since this is dependent on the agent's internal belief, we call this kind of exploration 'agent-centric'. In the next section we first consider a different source of signal which is environment-centric, and then discuss how it can be used to augment agent-centric exploration as well.

B. Environment Change

Seeing as how interesting manipulation behavior often involves changes in object states, and how this corresponds to change in visual features, we seek to autonomously estimate environment change from observed data. To capture environment interaction, the change metric should ignore differences in the robot's position, and only highlight movement of objects in the scene [27]. How then can we extract these ground truth change images from incoming image observations?

Our source of signal is assuming knowledge of the visual appearance of the robot, using which we train a segmentation model $m_\phi(.)$ to mask out the robot from the scene. Training this model is a one time cost, since the robot appearance is invariant across multiple tasks in the environment and even across different domains. We can use this model to measure the environment change f_c between a pair of image x_i, x_j:

$$\begin{aligned}
f_c(x_i, x_j) = f(&||m_\phi(x_i) - m_\phi(x_j)||_2, \\
&||\Psi(m_\phi(x_i)) - \Psi(m_\phi(x_j))||_2)
\end{aligned} \tag{2}$$

Here the heuristic function f takes into account pixel distance, blurring to remove shadows and relective surface artifacts, and Ψ denotes visual features from a pretrained segmentation network [37] (full details included in the appendix on our website), and returns a binary image indicating the pixels where change has been detected. We further apply a threshold for the change image, in order to minimize false detections. We don't require this change function to be fully accurate, and have found that our approach is robust to some error in the change image.

a) Environment-centric exploration

Using the norm of the change image as a metric, we can use off-policy RL [38], [39], [40] approaches to train a policy for control. The approach we use is to incorporate the metric into a world model by training the features s_t to also predict the change in the environment between observation o_t and the initial observation of the trajectory o_0, by adding an additional change predictor module $r_\theta(c_t|s_t)$. This is optimized by maximizing $\mathbb{E}[\log p(c_t|s_t)]$, similar to the image decoder, where c_t is the change image. While exploring under this objective, we optimize:

$$\arg\max_{a_1..a_T} \mathbb{E}_{s\sim\rho(s)}[\sum(r_\theta(c_{t+1}|s_{t+1})\big|s_t, a_t)] \tag{3}$$

b) Change-space Agent-centric exploration

Since the agent now models the environment change in its internal belief, it can leverage errors in this model to direct exploration. Just as previous exploration approaches maximize uncertainty of next state using the model [16], [5] the agent can maximize uncertainty over the *change* prediction. Thus, the agent will collect data that leads to information gain specifically about how the objects in the environment move, avoiding being stuck on gathering information pertaining to the robot's own body. Thus the agent will collect data that includes more information about object interactions. Specifically, we implement this by training an ensemble of models for $p(c_{t+1}|c_t, a_t)$, where c_t and a_t are the predicted change and action at time t respectively. To maximize uncertainty in change space, we optimize for actions that maximize the variance of the ensemble prediction (here s_t is a latent sampled from the world model) :

$$\arg\max_{a_1..a_T} \mathbb{E}_{s\sim\rho(s)}[\mathrm{Var}_k(p_{\phi^{(k)}}(r_\theta(c_{t+1}|s_{t+1}))\big|r_\theta(c_t|s_t), a_t)] \tag{4}$$

c) Control

Now that the features of the world model are trained to predict the environment change, we can explore by planning through the model adding the objectives from 4 and 3. We use the Cross entropy method [41] for planning, where we sample action proposals from an initial distribution, pick the top trajectories based on reward and refit the sampling distribution. Further, we train Advantage Weighted Regression (AWR) on the collected offline trajectories to maximize the environment change in the feature space of the world model. When sampling, given an observation, we first run the learned AWR policy through the model in an open-loop manner to get a sequence of actions. We use this as the mean of the initial normal sampling distribution for CEM, to bias the optimization procedure towards trajectories that are likely to have high environment change.

C. Leveraging Visual Priors

While environment change and ensemble disagreement can provide useful signal for driving behavior, the large work-spaces in the real world pose a major challenge for robots. Exploration methods often spend a lot of time in free space, and collect a large number of samples without interacting with any objects. This is undesirable since this data contributes little to learning manipulation skills. Our approach to avoiding this is to leverage visual priors from offline data, helping understand *what* to explore. One instance of this is to leverage object-detectors to initialize the robot near regions of interest. Recent models [33] are quite robust and can identify objects even in cluttered scenes. Using RGBD cameras and homography calibration for the robot with the cameras, we can then initialize the robot end effector close to the center of the object point-cloud, thus ensuring that data-collection is more likely to see object interactions. This approach does not preclude training on undetected objects, since the robot can always randomly sample points in the full workspace to initialize at later, and will likely be more proficient after it has learned skills efficiently on all the detected objects. For a image that has k detected masks $M_1, ..., M_k$, the robot can arbitrarily pick any mask for initialization every episode. However, in order to study exploration for independent objects separately, we enforce that the robot needs to reset to the same mask each time, and since this choice can be arbitrary, we also specify which mask should be selected, so that different methods can be evaluated on the same objects. We use the same visual prior for the baselines and ablations to make the exploration space feasible.

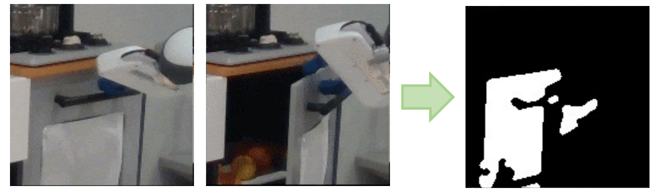


Fig. 4: An example of the change image extracted from a pair of images, as described in Equation 2. This is a binary image that detects pixels where change has occured.

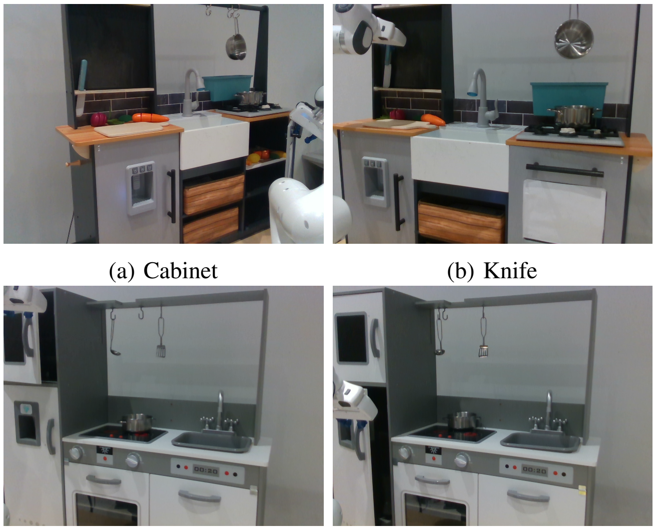
Fig. 5: We explore on 6 settings across two play kitchens. Top, from left: cabinet, knife, pan (kitchen1). Bottom, from left: top shelf, pot, fridge (kitchen2).

D. Achieving goals

Given the contact-rich data collected by the exploration controllers, how can we use this data to perform useful tasks? It is possible for the agent to sample goals from previously seen exploration data. Since the agent sees interesting data, any possible state can be a goal. Concretely, given some human sampled goal images, x_g, we leverage recent advances in goal-conditioned imitation learning, especially methods that leverage Nearest Neighbor-based techniques in a self-supervised representation space [42]. Our policy, π_{knn} scans through image features [43] in the exploratory data, and selects the top trajectory matches shown in Equation 5.

$$\tau^\star = \mathrm{argmin}_i \min_{x_j \in \tau_i} ||\phi(x_g) - \phi(x_j)||_2 \tag{5}$$

Since our method has seen interesting trajectories, it is more likely to see semantically useful goals, and thus when a human provided goal z_{gh} is given, more likely to reach it.



(a) Cabinet (b) Knife

(c) Top Shelf (d) Fridge

Fig. 6: Manually specified goals used for zero-shot evaluation, after the completion of the exploration phase.

V. Experimental Setup

In our experiments, we seek to address the following questions : 1) Does our system enable autonomous exploration leading to the discovery of interesting states in complex real world environments ? 2) How does the quality of this data compare to that of current SOTA approaches? 3) Is it possible to use this data to reach human specified goals to perform useful tasks?

a) Real World Setup

We tested our system on a Franka Panda 7dof robot, and on two different real-world kitchen play-sets, which have many diverse objects and possible manipulation tasks, comprising a very large search space (both are about 100cm X 100cm X 100cm). Specifically, we investigate 6 object regions across two kitchens detected by our visual prior approach [33], as shown in Figure 3. Namely, these are the knife, cabinet and the hanging pan from the first kitchen, and the top shelf, fridge and pot from the second kitchen (Figure 5). During training we provide minimal resets via human intervention, and only when the object is in an unresettable state (for example when the knife or pan has fallen down), or for safety reasons. Our setup uses 2 cameras to cover the entire scene, and the observation space consists of a single 128X128 size RGB image from the camera that is farther from the robot end effector, which provides a more complete view of interaction. We execute 6-dof control on the arm along with open-close gripper action. The change image is resized to a 32X32 binary image for prediction. Training and sampling are run asynchronously to maximize efficiency of the system.

b) Training Procedure

For each of the regions, we first collect a random dataset of 25 trajectories. Each of these as well as the subsequent collected trajectories are 20 timesteps long. The world models in all methods use an RSSM [17], and the image encoders and decoders use the NVAE architecture [30]. To extact the environment centric metric, we train a Mask RCNN model [37] on 200 instances of the robot. We extract features after applying this mask to make them agent agnostic, and train a joint segmentation model using data from both play kitchens.

c) Baselines and Ablations

We compare against `LEXA`[6], a state-of-the art self-supervised exploration approach for continous control in manipulation settings. `LEXA` outperforms various other self-supervised approaches, such as SkewFit[24], DIAYN[19], Dynamic Distance Learning [44] on a complex simulated kitchen environment both in terms of the exploratory data seen, and the success rate of reaching discovered goal images. We provide this baseline with the same world model architecture as `ALAN`. Next, we ablate the need of our agent-centric module, which explores in the change space. This is to test our hypothesis that the robot should continually collect data where the model predictions regarding environment change are inaccurate. We test if this ability is crucial, by running the environment-centric exploration model, which only uses the intrinsic reward described in Equation 2. We run two versions of this, `EC` which uses the model for planning, and `AWR` which just uses the trained AWR policy, without planning.

VI. Results

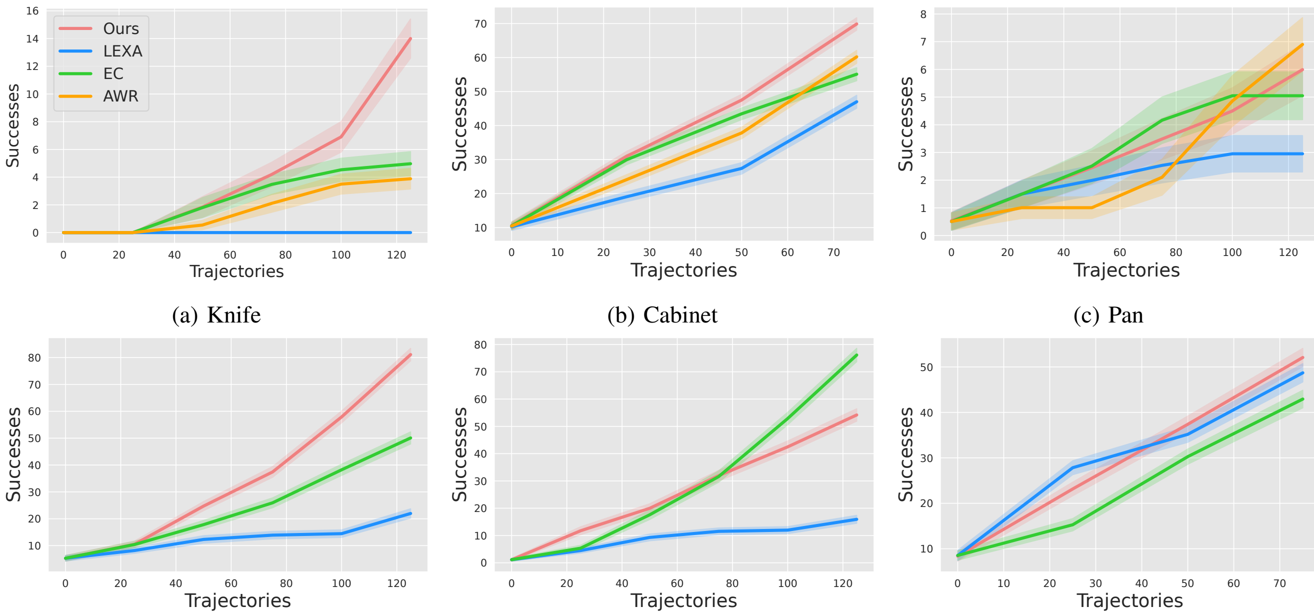
a) Exploration

We need a metric to evaluate the quality of the exploration data. While the change image norm is a good proxy for measuring object interaction, it does not consider if the different states are semantically interesting. Thus we define an exploration metric that measures the *number* of successful interactions over time, which are are determined by a human operator. We now describe how we define success for a trajectory, for each of the tasks considered -

- Cabinet, fridge, shelf doors - has been opened or closed
- Knife - lifted up
- Pan - unhooked, fully removed from hanger
- Pot - pushed/lifted/knocked over

We have included examples of successful exploration trajectories on our website for clarity. Using this success criteria, we present evaluation of the exploratory data collected, in Figure 7. For each task we run about 100-150 trajectories, and plot the cumulative number of successful exploration trajectories against the total number of trajectories seen during the exploration phase.

We can see that `ALAN` (red) outperforms or matches the exploration success of all the other approaches in five out of six tasks, and also sees large number of successes for the top shelf. Further, we see that just maximizing the environment-change metric using `EC` or `AWR` leads to much better performance than LEXA, the previous state-of-the-art self-supervised exploration approach. We find that because the robot arm takes up a large portion of the observation, `LEXA` tries to collect data to resolve modelling inaccuracies of the arm. This is especially the case for tasks where random interactions are less likely to produce significant changes in the object, such as the particularly challenging knife task where `LEXA` never sees the picking up behavior. Further we see that on this task, having the agent-centric module



(a) Knife (b) Cabinet (c) Pan

(d) Fridge (e) Top Shelf (f) Stovepot

Fig. 7: Coincidental success for exploration on our six tasks, where the robot reaches a semantically meaningful state while collecting data during exploration. We can see that `ALAN` performs consistently well across tasks, and that just maximizing the change metric `AWR`, `EC` also yields much better data than previous state of the art approach `LEXA`.

which maximizes uncertainty in change space significantly improves performance over `EC` and `AWR`. For tasks like the top shelf which require less precise control, simply maximizing environment change is sufficient to collect high-quality data. However, even with slightly more involved control, such as the fridge task which requires the same object motion but has the robot in a more constrained position, addressing modelling inaccuracies in the change prediction is more critical. Moreover, using the agent-centric module leads to more robust performance for goal reaching, as described in the next section.

b) Achieving Goals

Given the exploration data collected, can it be used to perform useful human specified tasks ? For this, we use the nearest-neighbor (kNN) approach outlined in section IV-D, paired with model-based refinement to reach different human-specified goals. Specifically, once the kNN approach finds a trajectory, we use the action sequence as the mean of the initial sampling distribution of the CEM optimizer. The goals consist of a fully open fridge, cabinet or shelf, and a picked-up knife, as shown in Figure 6. Since `AWR` has almost identical results for exploration and goal-reaching to `EC` on the first kitchen, and since they both optimize the same objective, we did not run it on the second kitchen (and therefore for the fridge and top shelf tasks). For each task, we run kNN on the exploratory data, in a visual feature space [43] and select the best trajectory to execute conditioned on the start and goal images. We execute the top two trajectories five times each, collecting 10 different trials. We present average success rates in Table I. We can see that our approach performs consistently well across tasks. Without the agent-centric module, there is no success on the difficult knife task, and overall performance across the remaining tasks is worse in terms of robustness. Moreover these results demonstrate the effectiveness of the environment change metric, since `LEXA` shows no success for three of the four tasks. This confirms our hypothesis that better exploratory data leads to better goal-reaching ability, and that leveraging both environment and agent centric objectives leads to best results.

	Cabinet	**Knife**	**Fridge**	**Top Shelf**
`LEXA` [6]	0.20	0.00	0.00	0.00
`EC`	0.70	0.00	0.50	**0.90**
`AWR` [38]	0.50	0.00	-	-
`ALAN` (ours)	**1.00**	**0.60**	**0.70**	0.80

TABLE I: Success rate for goal reaching. `ALAN` is the only approach to get success on the challenging knife pick-up task, and just maximizing change (`EC`) also leads to much stronger performance than `LEXA`.

VII. Discussion and Limitations

We present `ALAN`, an autonomously exploring agent that can efficiently explore in challenging real world environments. Our approach computes change in the environment, and utilizes it both directly as an environment-centric signal, as well as modelling the change and taking actions that maximize uncertainty in change space, which provides agent-centric signal. This reward in the absence of true task rewards helps our agent autonomously discover manipulation skills and perform useful tasks without any supervision. In the future, we hope to investigate distilling exploration data into a general goal-achieving policy, and studying continual learning across different tasks using a joint world model.

REFERENCES

[1] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," *JMLR*, 2016.

[2] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke *et al.*, "Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation," *arXiv preprint arXiv:1806.10293*, 2018.

[3] D. Ha and J. Schmidhuber, "World models," *arXiv preprint arXiv:1803.10122*, 2018.

[4] D. Hafner, T. Lillicrap, J. Ba, and M. Norouzi, "Dream to control: Learning behaviors by latent imagination," *arXiv preprint arXiv:1912.01603*, 2019.

[5] R. Sekar, O. Rybkin, K. Daniilidis, P. Abbeel, D. Hafner, and D. Pathak, "Planning to explore via self-supervised world models," *ICML*, 2020.

[6] R. Mendonca, O. Rybkin, K. Daniilidis, D. Hafner, and D. Pathak, "Discovering and achieving goals via world models," *NeurIPS*, 2021.

[7] A. Strehl and M. Littman, "An analysis of model-based interval estimation for markov decision processes." *Journal of Computer and System Sciences*, 2008.

[8] M. O. Duff and A. Barto, "Optimal learning: Computational procedures for bayes-adaptive markov decision processes," Ph.D. dissertation, University of Massachusetts at Amherst, 2002.

[9] P. Poupart, N. Vlassis, J. Hoey, and K. Regan, "An analytic solution to discrete bayesian reinforcement learning," in *ICML*, 2006.

[10] M. Bellemare, S. Srinivasan, G. Ostrovski, T. Schaul, D. Saxton, and R. Munos, "Unifying count-based exploration and intrinsic motivation," in *Advances in Neural Information Processing Systems*, 2016, pp. 1471–1479.

[11] G. Ostrovski, M. G. Bellemare, A. Oord, and R. Munos, "Count-based exploration with neural density models," in *International conference on machine learning*. PMLR, 2017, pp. 2721–2730.

[12] H. Tang, R. Houthooft, D. Foote, A. Stooke, O. Xi Chen, Y. Duan, J. Schulman, F. DeTurck, and P. Abbeel, "# exploration: A study of count-based exploration for deep reinforcement learning," *Advances in neural information processing systems*, vol. 30, 2017.

[13] D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell, "Curiosity-driven exploration by self-supervised prediction," in *ICML*, 2017.

[14] K. Lowrey, A. Rajeswaran, S. Kakade, E. Todorov, and I. Mordatch, "Plan online, learn offline: Efficient learning and exploration via model-based control," *arXiv preprint arXiv:1811.01848*, 2018.

[15] I. Osband, J. Aslanides, and A. Cassirer, "Randomized prior functions for deep reinforcement learning," in *Advances in Neural Information Processing Systems*, 2018, pp. 8617–8629.

[16] D. Pathak, D. Gandhi, and A. Gupta, "Self-supervised exploration via disagreement," in *International Conference on Machine Learning*, 2019, pp. 5062–5071.

[17] D. Hafner, T. Lillicrap, I. Fischer, R. Villegas, D. Ha, H. Lee, and J. Davidson, "Learning latent dynamics for planning from pixels," *arXiv preprint arXiv:1811.04551*, 2018.

[18] S. Levine and V. Koltun, "Guided policy search," in *ICML*, 2013.

[19] B. Eysenbach, A. Gupta, J. Ibarz, and S. Levine, "Diversity is all you need: Learning skills without a reward function," *arXiv preprint arXiv:1802.06070*, 2018.

[20] A. Sharma, S. Gu, S. Levine, V. Kumar, and K. Hausman, "Dynamics-aware unsupervised discovery of skills," *arXiv preprint arXiv:1907.01657*, 2019.

[21] D. Warde-Farley, T. Van de Wiele, T. Kulkarni, C. Ionescu, S. Hansen, and V. Mnih, "Unsupervised control through non-parametric discriminative rewards," *arXiv preprint arXiv:1811.11359*, 2018.

[22] M. Andrychowicz, F. Wolski, A. Ray, J. Schneider, R. Fong, P. Welinder, B. McGrew, J. Tobin, P. Abbeel, and W. Zaremba, "Hindsight experience replay," *arXiv preprint arXiv:1707.01495*, 2017.

[23] A. V. Nair, V. Pong, M. Dalal, S. Bahl, S. Lin, and S. Levine, "Visual reinforcement learning with imagined goals," in *NeurIPS*, 2018, pp. 9191–9200.

[24] V. H. Pong, M. Dalal, S. Lin, A. Nair, S. Bahl, and S. Levine, "Skew-fit: State-covering self-supervised reinforcement learning," *arXiv preprint arXiv:1903.03698*, 2019.

[25] Y. Zhang, P. Abbeel, and L. Pinto, "Automatic curriculum learning through value disagreement," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[26] L. Shao, T. Migimatsu, Q. Zhang, K. Yang, and J. Bohg, "Concept2robot: Learning manipulation concepts from instructions and human demonstrations," *The International Journal of Robotics Research*, vol. 40, no. 12-14, 2021.

[27] S. Bahl, A. Gupta, and D. Pathak, "Human-to-robot imitation in the wild," in *RSS*, 2022.

[28] A. S. Chen, S. Nair, and C. Finn, "Learning generalizable robotic reward functions from" in-the-wild" human videos," *arXiv preprint arXiv:2103.16817*, 2021.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[30] A. Vahdat and J. Kautz, "Nvae: A deep hierarchical variational autoencoder," *Advances in Neural Information Processing Systems*, vol. 33, pp. 19 667–19 679, 2020.

[31] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," *arXiv preprint arXiv:1401.4082*, 2014.

[32] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[33] X. Zhou, R. Girdhar, A. Joulin, P. Krähenbühl, and I. Misra, "Detecting twenty-thousand classes using image-level supervision," *arXiv preprint arXiv:2201.02605*, 2022.

[34] J. Schmidhuber, "A possibility for implementing curiosity and boredom in model-building neural controllers," 1991.

[35] ——, "Curious model-building control systems," in *[Proceedings] 1991 IEEE International Joint Conference on Neural Networks*. IEEE, 1991, pp. 1458–1463.

[36] D. Hafner, T. Lillicrap, M. Norouzi, and J. Ba, "Mastering atari with discrete world models," *arXiv preprint arXiv:2010.02193*, 2020.

[37] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *ICCV*, 2017.

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

[39] A. Nair, A. Gupta, M. Dalal, and S. Levine, "Awac: Accelerating online reinforcement learning with offline datasets," *arXiv preprint arXiv:2006.09359*, 2020.

[40] I. Kostrikov, A. Nair, and S. Levine, "Offline reinforcement learning with implicit q-learning," *arXiv preprint arXiv:2110.06169*, 2021.

[41] R. Y. Rubinstein, "Optimization of computer simulation models with rare events," *European Journal of Operational Research*, vol. 99, no. 1, pp. 89–112, 1997.

[42] J. Pari, N. Muhammad, S. P. Arunachalam, L. Pinto *et al.*, "The surprising effectiveness of representation learning for visual imitation," *arXiv preprint arXiv:2112.01511*, 2021.

[43] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta, "R3m: A universal visual representation for robot manipulation," *arXiv preprint arXiv:2203.12601*, 2022.

[44] K. Hartikainen, X. Geng, T. Haarnoja, and S. Levine, "Dynamical distance learning for semi-supervised and unsupervised skill discovery," *arXiv preprint arXiv:1907.08225*, 2019.