ALAN: Autonomously Exploring Robotic Agents in the Real World

Anonymous Author(s)
Affiliation
Address
email

Abstract: In order to build robotic agents that can autonomously operate in the real world, it is crucial to explore the environment. While it is possible to build agents that can learn 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. Our approach builds a shared model of the world, across all possible tasks, and continuously explores. We propose a novel intrinsic motivation reward that leverages both self-supervised agent-centric prediction as well environment-centric priors for visual change. We show our approach on a real world play kitchen setting, performing multiple manipulation tasks and showing much better exploratory performance than state-of-the-art methods. Videos can be found at https://robo-explorer.github.io/

Keywords: Exploration, World Models, Learning Control

1 Introduction

To gain true autonomy, 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, then they could explore their environments and learn interesting and useful 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 observed images from diverse domains, thus enabling reuse of data for learning useful behaviors. 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]. 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 is very expensive. 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. We note that this is an 'environment-centric' reward in contrast to maximizing model uncertainty which is more 'agent-centric', since it is concerned with controlling predictions in its own mental model. Furthermore, the environment change is a task-agnostic objective, and will generally guide the robot towards more contact-rich behavior, which is essential for learning manipulation tasks. Using the world model, the agent can learn to predict what the environment change will be in

Submitted to the 6th Conference on Robot Learning (CoRL 2022). Do not distribute.

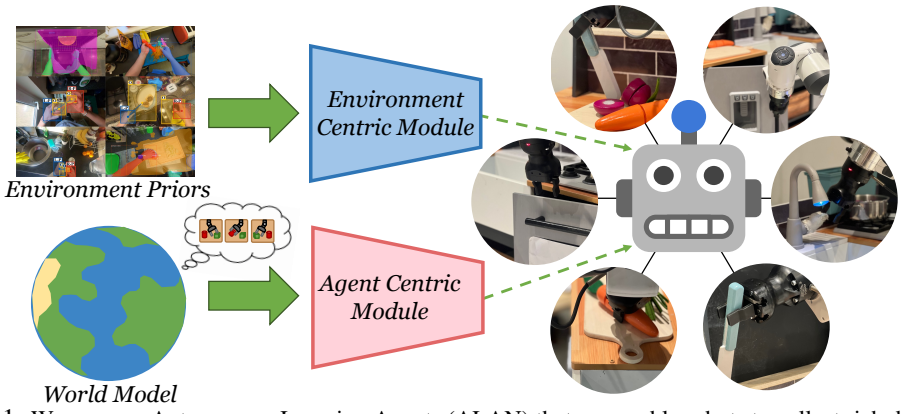


Figure 1: We propose Autonomous Learning Agents (ALAN) that can enable robots to collect rich data from their environment efficiently. The agent autonomously identifies workspace of interest, and learns a joint world model on all data. ALAN follows two intrinsically defined objectives - taking actions that cause difficulty for prediction (the agent centric module), and actions driven by environmental priors (environment centric module).

the low-dimensional representation space, enabling these agents to explicitly optimize for this metric when planning, leading to trajectories that interact with objects.

Our contributions can be summarized by the following. We present ALAN, an efficient real world exploration algorithm with a novel intrinsic motivation reward that extracts better signal from the environment by learning to predict which transitions cause large visual change. Our approach shows strong exploratory performance and goal reaching on several manipulation tasks in a real world toy kitchen, that outperform SOTA baselines. We also show that the data is useful for reaching human-specified goals by training distilling the collected experience into a goal-conditioned policy.

2 Related Work

Exploration In reinforcement learning (RL), exploration has been studied in various different contexts ranging from highly tabular to complex continuous spaces. For discrete settings, the concept of exploration can be analytically defined as state visitation counts [7] or as probability distributions over visited states [8, 9]. For complex states, such as images, previous works have used neural networks to approximate state counts [10]. A common way to discribe an intrinsic reward for exploration is to use either the error [11] or uncertainty [12, 13] in prediction about how the environment and agent would interact. Pathak et al. [14] propose a differentiable intrinsic reward which measures the disagreement between an ensemble of models via the variance of the set. Sekar et al. [5] leverage a disagreement intrinsic reward, but explore in an imagined space via a world model [15, 4, 16]. These methods also do not scale to real world settings. Prior work has attempted to have the agent provide its own supervision in the real world [17, 18, 19], however, these approaches only trained for specific tasks and cannot share data between each one. Most of these approaches [20, 2] suffer from sample efficiency. Our approach leverages both the agent's own supervision, is able to share data across manipulation tasks and leverages an initialization from human behavior priors.

Self-Supervised Goal Reaching Collecting good data and learning skills is important, however, an agent needs to be able to distill this experience into a policy that is able to reach some form of goals. Traditionally, prior work has learnt a policy that directly practices on multiple tasks, and conditions on some provided goal. This goal can come from supervision [21, 22] or can come from the agent's own experience [23]. Training an agent in the real world requires operating from raw sensory data, and being able to achieve goals specified by images only. Prior work has used a mix of visual representation such as constrastive [24] or generative models [25, 17]. While visual representations can provide some signal, using simple distance metric does not enable exploration of the environment. Pong et al. [25] and Zhang et al. [26] employ goal proposal mechanism trained on previously seen states. Proposing goals close to previously achieved ones tends to keep the agent stuck performing the same types of task, and does not allow for learning of complex behaviors. Mendonca et al. [6] propose an exploration approach that is able to leverage an imagined world model to learn to achieve

sampled goals. One of our key insights is to utilize a world model and practice goal reaching in imagination, similarly to [6]. However, it is difficult to know *what* to focus on while exploring. For this, we use offline priors from human behavior data from the internet.

3 Background

Model-Based RL and Planning Model Based RL operates in a Markov Decision Process (MDP). An 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 $\mathcal{R}(s_t, a_t)$. Specifically, 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 $\mathcal{T}$. 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.

Sample Efficient Model-Based RL In order to do as efficient model-driven learning as possible, 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 artefacts. Thus, we employ the Recurrent State-Space Model (RSSM) from Hafner et al. [4]. Given an image input at timestep t: x_t, we obtain a low dimensional embedding $e_t = E_\phi(x_t)$, where E is the ResNet-based [27] encoder from Vahdat and Kautz [28]. 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 $p_\phi(x_t|s_t)$. Throughout the paper we refer to the RSSM as a world model, and we train it using the ELBO loss [29, 30].

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, Pathak et al. [11] use 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 driven approach to optimizer it. Thus, Pathak et al. [14] propose to minimize the *disagreement* between an ensemble of dynamics model $f_{\phi^{(k)}}$ for $k = 1, ..., M$. The disagreement reward can be described as

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

4 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 experience with real hardware.

Figure 2: Visualizations of the object detections we use [31]. We sample masks for each workspace.

To this end, we propose ALAN, an autonomous robot learning algorithm that is able to efficiently explore in the real world, and achieve various goals, which can be self-supervised or human provided. Firstly, we use offline visual data to reduce the search space for the robot. ALAN identifies the locations of potential interesting and complex interactions for the robot. Secondly, 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. We explain how these insights enable efficient autonomous learning below.

4.1 Priors from offline data

In any given real world environment, there are many interesting aspects to manipulate. However, there are even more parts of the environment which are akin to free space, and do not provide much insight to the robot. Exploration methods often get stuck in "free space" with the agent moving itself only. One such way to avoid this is to leverage visual priors from offline data, helping understand *what* to explore. One instantiation of this is using object-detectors. Recent models [31] are robust and can identify objects even in cluttered scenes . We can use these to initialize the robot close to the object. Consider the scene image at time t, I_t. We process the image using a visual object detector O_ϕ and obtain k masks $M_1, ..., M_k$. At training time, we sample M_i, obtain the image center of the mask (u_i, v_i), and convert using an approximate transform (R, t) obtained from calibration. We obtain a trajectory τ from the central reset pose p_0 to p_i, which is the pose of the centre of the mask in the world coordinate frame. We use an off-the-shelf RRT* [32] planner to reach these locations.

4.2 World Model

Traidional RL methods struggle to train in the real world while interacting with many different tasks due to sample efficiency. To overcome this, we enable ALAN to leverage *all* the data seen for every task. Naively combining all the data the robot sees will not work, as it will override previous experience. Thus, we leverage the RSSM model [4, 16, 6, 5] discussed above. Since our inputs are high dimensional, we augment the RSSM to learn the agent learns a compact representations of its environment by learning to predict observed data trajectories using the RSSM model ([4]), which has recurrence to model long-range dependencies and also accounts for the stochasticity of the environment. More concretely, this *world model*, W_θ can be described using the following components.

Image Encoder	$h_t = [e_t^1, e_t^2, e_t^3], e_t^i = \phi_i(xi_t)$	Dynamics Prior	$p_\theta(s_{t+1}\|s_t, a_t)$
Image Decoder	$f_\theta(x_t^i\|s_t), i = 1, 2, 3$	Dynamics Posterior	$q_\theta(s_{t+1}\|s_t, a_t, h_{t+1})$
Embed Decoder	$g_\theta(e_t^i\|s_t), i = 1, 2, 3$	Change Predictor	$r_\theta(c_t^i\|e_t^i, e_{t-1}^i)$

(1)

Our Image Encoder and Decoder employ the visual encoding approach from Vahdat and Kautz [33], while the rest of our world model follows a similar structure to [4]. The use of the world model enables our approach to not only work for combining data from many tasks but it also allows for efficient exploration, as we can use the model think about where to in fact explore. We call this *prospective agent-centric exploration*.

4.3 Prospective Agent-Centric Exploration

When interacting with the world, the robot must carefully decide how to spend its resources. Knowing in advance about what interactions might provide some benefit to the robot is very important. The robot can directly leverage the world model, W_θ for this. Given some initial observation x_1, the robot could sample actions $a_1, ..., a_t$ that it thinks might be interesting. Concretely, the robot defines an *intrinsic* motivation score, r_t^i, as the robot does not have any other external reward. This is derived from function $I(.)$ for any transition that it images z_t, a_t, z_{t+1}. We sample many such action sequences, $\mathcal{A}_1, ..., \mathcal{A}_M$ and run them through the intrinsic function. $W_\theta(x_1, \mathcal{A}_m)$ describes the output of the world model $z_1, z_2, ..., z_T$, all states in imagination. The goal for the agent is to maximize:

$$\mathcal{A}^\star = \text{argmax}_m I(W_\theta(x_0, \mathcal{A}_m)) \tag{2}$$

An important aspect to our exploration approach is to define the agent-centric exploration metric as the *disagreement* between a set of predictive models $f_{\phi^{(k)}}$, as explained in the previous section:

$$\mathbb{E}_{z\sim\rho(z)}[\text{Var}_k(f_{\phi^{(k)}}(z_{t+1}|z_t, a_t)] \tag{3}$$

Algorithm 1 Autonomous Learning Agent (ALAN)

Run visual processing (eg: object detectors) to get workspaces $\mathcal{T}_1, \mathcal{T}_1$ .. $\mathcal{T}_n$.
Collect random trajectories for each workspace $\mathcal{T}_i$
Train a joint World Model $\mathcal{W}$ on all the aggregated data
while exploring **do**
  Randomly select workspace $\mathcal{T}_i$ to explore.
  Initialize workspace specific model $\mathcal{W}_i = \mathcal{W}$ and workspace specific data $\mathcal{D}_i$
  while number of new trajectories is less than N **do**
    Train $\mathcal{W}_i$ on $\mathcal{D}_i$.
    Run CEM with model $\mathcal{W}_i$ to maximize intrinsic objective 5
  end while
  Add $\mathcal{D}_i$ to $\mathcal{D}$
  Train joint model $\mathcal{W}$ on aggregated data $\mathcal{D}$
end while

Here, z_t is sampled fro the RSSM (W_θ) described previously. ALAN iterates over these intrinsic rewards for action sequences $\mathcal{A}$ in imagination, in a similar fashion to the CEM algorithm. We use this intrinsic disagreement objective as it allows the robot to explore skills. While this will eventually learning meaningful skills, the robot will be curous about spurious actions. To avoid this, it is important for the exploration to be grounded in the *environment*.

4.4 Environment-Centric Exploration

Grounding the robot with how the environment behaves in important to guide exploration. No matter the skills of the robot, it should always be exploring some important aspects of the environment. We denote this as Environment-Centric exploration. This often involves interactions with objects which cause changes in the environment [34, 35]. Given a scene observation at x_t, we can compute this using visual features Ψ, for instance from a pretrained visual network. Specifically we use a semantic segmentation network [36]. We define a function f_c that measures the *change* in the environment.

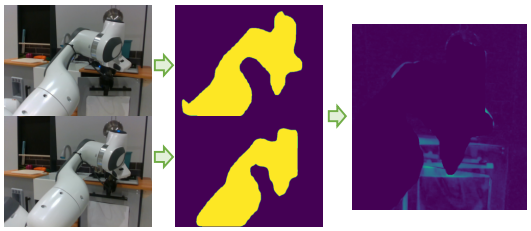


Figure 3: We present a visualization of our environment centric metric, described in (4), right shows the change metric between the two images.

$$f_c(x_t, x_{t+k}) = f_{\text{heuristic}}(x_t, x_{t+k}) + ||\Psi(x_t) - \Psi(x_{t+k})||_2 \tag{4}$$

Here, the first term is a heuristic image change function, and $\Psi(x)$ are the semantic segmentation features. This can be used an exploration bonus to the intrinsic disagreement approach described. However, this change can only be measured on collected trajectories, which means that we cannot imagine potentially environment-changing action sequences. In order to reconcile this, we distill a network r_θ fits world model image embeddings e_t of images x_t to $f_c(x_t, x_{t-1}) = c_t$. This change predictor network's loss can be described as the $\mathcal{L}_c = ||f_c(x_t, x_{t-1}) - r_\theta(c_t|e_t, e_{t-1})||_2$.

This gives rise to our intrinsic objective used for classifying action sequences $a_{1:T} = \mathcal{A}$, $I_\theta(.)$:

$$I_\theta(\mathcal{A}) = \sum_{t=1}^{T} \mathbb{E}_{z\sim\rho(z_t)}[\text{Var}_k(f_{\phi^{(k)}}(z_{t+1}|z_t, a_t)] + r_\theta(c_t|e_t, e_{t-1}) \tag{5}$$

4.5 Continual Exploration and Goal Reaching

We summarize our approach, ALAN, in this section. The first part of the approach, involves collecting exploration experience using our world model W_θ. We assume that this has been on trained on a little bit of data so that the predictions are grounded in the environment. We use the sampling approach described in Equation 2 with ranking function that uses our agent and environment centric metric

Figure 4: We explore on 6 real world settings as pictured above (left): fridge, tap, knife, stovepot, vegetables and cabinet. On the right, we present a full view of all the worksapces [37].

(Equation 5). The robot collects exploration data on each region of the environment, $\mathcal{T}_i$ and shares the data between them.

An important aspect to our approach is how to train on one region, $\mathcal{T}_i$ without confusing it with another region of the workspace $\mathcal{T}_j$. ALAN is able to continuously train, using data from previous experiences. Specifically, we propose always tuning the world model, W_θ on all the data, but when the robot explores the workspace $\mathcal{T}_i$, we propose only finetuning on data from that region. This allows for the model to focus on a task, and yet aggregate experiences.

Achieving goals Given the contact-rich data collected by the exploration controllers, the question is how can we efficiently and effectively use this data to learn manipulation skills? Since there is no provided objective, the agent will have to set its own goals from the collected experience, and practice reaching them. 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 train a goal conditioned policy π_{GC} that can learn from the exploratory data. The policy is trained on world model embeddings z_t, and a randomly sampled goal from the same trajectory z_g. The policy predicts the action a_t taken at timestep t.

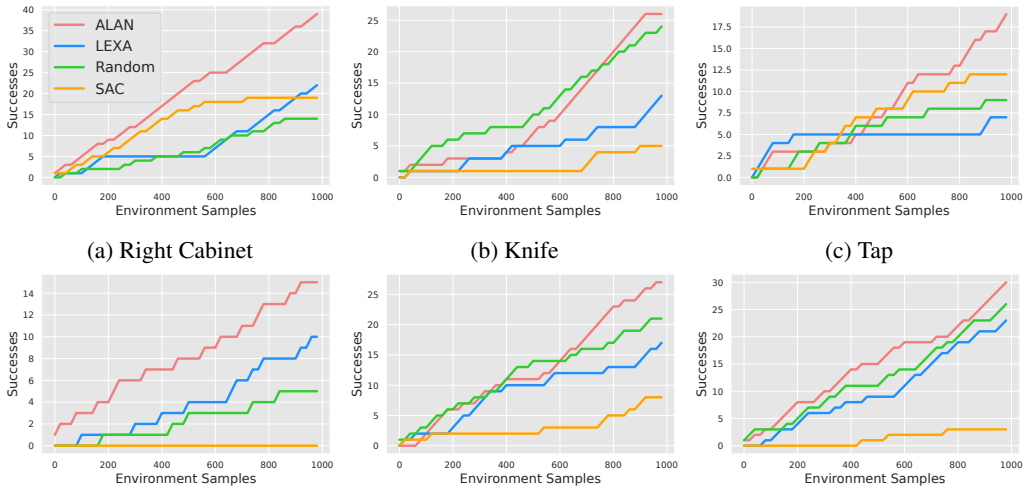
$$\pi^\star = \text{argmin} \sum_{i,j} ||\pi_{GC}(z_i|z_j) - a_j||_2 \tag{6}$$

Since our method has seen interesting trajectories, it is more likely to see diverse goals, and thus when a human provided goal z_{gh} is given, more likely to reach it. We find that ALAN provides just this, as the exploration of our method is meaningful and leads to a lot of rich data.

5 Experimental Setup

In our experiments, we seek to address the following questions : 1) Does our system enable autonomous exploration and discovery of useful manipulation skills in a complex real world environment ? 2) How does the quality of this data compare to prior approached? 3) Is it possible to use this data to reach human specified goals?

Real World Setup We tested our system on a Franka Panda 7dof robot, and on a real-world kitchen play-set (taken from the bridge dataset [37]), which has many diverse objects and possible manipulation tasks, comprising a very large search space (112cm X 92cm X 110cm). Specifically, the 6 object regions that are returned by our human prior, the object detector from Zhou et al. [31]. These regions were chosen based on high confidence from the model (Figure 2). The following locations are sampled: the right cabinet, burner knobs, stove pot, tap, knife, vegetables on the right counter-top, and the left cabinet. We sample different meaningful goal images for each of these regions. For example, this involves lifting the knife up in the air, removing the pot, opening or closing the cabinet.



(a) Right Cabinet (b) Knife (c) Tap

(d) Left Cabinet (e) Vegetables (f) Stovepot

Figure 5: Coincidental success for exploration on our six workspaces. We can see that ALAN sees a lot more success in every task compared to baseline exploration approaches.

Training Procedure For each of the regions, we first collect a random dataset of 50 trajectories. We then train a joint world model on these W_θ. As described above, ALAN trains the joint world model on all worskpaces continually, but when exploring an area it only uses data from that region. The process of data collection takes about 45 seconds per trajectory. The finetuning process for W_θ takes about 20 minutes. Our method uses the NVAE [28] architecture, as well as the RSSM from Mendonca et al. [6]. To extract the environment centric metric, we train a Mask RCNN model [36] on 200 instances of the robot. We extract features after applying this mask to make them agent agnostic.

Baselines and Ablations It is important to test the different design decisions and choices made in designing ALAN. Firstly, we test if our novel exploration metric, which combines both agent and envrionment centric exploration, works better than exploration approaches that just use an agent centric approach, such as Mendonca et al. [6]. This method is one of our baselines, which we call `LEXA`. It uses the same approach and training procedure as ALAN, however, does not use our environment centric intrinsic reward described in Equation 4, and only uses Equation 3 as a motivation score. Secondly, we compare whether the world model, W_θ is useful. We call this method, `Rand`, as it collects random data only, this method uses a similar goal reaching approach as Pari et al. [38]. Finally, we test our method against state of the art RL approach, Soft Actor-Critic (SAC) [39]. This approach using Q-Learning, but with our environment centric reward (Equation 4). The goal achieving mechanism for all is the same goal conditioned policy learning procedure described in the section above (Equation 6). We also test the goal conditioned policy against an RL policy trained in the world model's imagination, similar to Mendonca et al. [6].

6 Results

Exploration We present results of the exploratory data collected for ALAN, in Figure 5. It is important to measure the quality of the data collected. To this end, we define an exploration metric that measures the *number* of successful interactions over time. The x-axis in Figure 5 is the number of environment samples taken, and the y-axis is the number of successes. We can see that in all of the tasks, our approach (red) outperforms the baseline approaches by a significant margin. Empirically, we have observed this as well, as we have seen the robot interact a lot more the workspace in meaningful ways. LEXA [6], which does not use our environment-centric intrinsic module tends to perform worse when more precise control is required, for example when handling the knife or the tap. On the other hand, these are the tasks where Random tends to perform better. We see that overall,

Figure 6: Manually specified goals used for evaluation

apart from ALAN, SAC [39] tends to perform better than the baseline. This shows that latching on to our intrinisc motivation (visual change) objective does lead to success.

Additionally, we see that ALAN continuously improves as it sees more data. Towards the end of the exploration process, it is almost constantly seeing successes. This shows that ALAN is in fact getting better with more exploration data. We suspect running our approach for longer will lead to even better data across all the tasks. This does not hold true for other methods, including LEXA, which follows a similar data aggregation and word model scheme as us.

Achieving Goals It is important to show that the exploratory data can be used for performing human specified tasks. For this, we use goal conditioned supervised learning to learn a goal reaching policy from the collected data, and also compare to an approach that trains a policy using model imagination (LEXA). We try 10 different trials for the goals shown in 6, and present the average success rates in Table 1. We can see that our approach significantly outperforms the random data baseline, while being comparable or better than SAC and LEXA. The importance of the world model is apparent from the cabinet opening task, as the approaches which do not use it fail to get any success. The difference between the tap goals is that goal1 requires more precise control as the handle is not pushed all the way down. Our approach is the only one that gets high success on this task, while remaining competitive on the other tap goal as well.

	Cabinet Open	Tap Goal 1	Tap Goal 2
No World Model:			
`Random` [38]	0.0	0.0	0.4
`SAC` [39]	0.0	0.6	0.4
No Environment-Centric Reward:			
`LEXA` [6]	0.4	0.2	1.0
`Disag + gcsl`	1.0	0.1	0.7
`ALAN` **(ours)**	1.0	0.9	0.5

Table 1: We present the evaluation (success rate) of the goal reaching experiments for two of our tasks.

7 Discussion and Limitations

We present ALAN, an autonomously exploring agent that can aggregate data and experience across multiple tasks. Our approach hinges on two major components, world-model based agent-centric intrsic motivation as well as environment-centric visual change. This reward in the absence of true task rewards helps our agent achieve success in the real world. We also show that for a subset of tasks, it is possible to distill this experience into a goal-conditioned reaching policy.

Limitations While enabling efficient exploration, our approach is centered on environment driven priors. Firstly, if the object detections are wrong or there are inaccuracies in the motion planning or calibration, this will lead to a bad initialization for exploration. This can be prevented by training a more robust policy with a larger workspace. Secondly, our approach hinges on the quality of the world model. When in the same kitchen, this model is able to aggregate data, however, this may not hold when there is too much variation between scenes or environments. Finally, our approach is limited by the quality of the visual change. If the visual change detects spurious correlations (such as lighting, artefacts etc), then it will focus on the wrong things. We saw to some extent, for example, the change metric focusing on shadows or lighting conditions at some points in time. In the future, we hope to address these limitations of robustness, diversity in the environments and reliability of the visual change metric.

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*. PhD thesis, 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*, pages 1471–1479, 2016.

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

[12] 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.

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

[14] D. Pathak, D. Gandhi, and A. Gupta. Self-supervised exploration via disagreement. In *International Conference on Machine Learning*, pages 5062–5071, 2019.

[15] 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.

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

[17] A. V. Nair, V. Pong, M. Dalal, S. Bahl, S. Lin, and S. Levine. Visual reinforcement learning with imagined goals. In *NeurIPS*, pages 9191–9200, 2018.

[18] D. Pathak, P. Mahmoudieh, G. Luo, P. Agrawal, D. Chen, Y. Shentu, E. Shelhamer, J. Malik, A. A. Efros, and T. Darrell. Zero-shot visual imitation. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 2050–2053, 2018.

[19] L. Pinto and A. Gupta. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. *ICRA*, 2016.

[20] H. Zhu, J. Yu, A. Gupta, D. Shah, K. Hartikainen, A. Singh, V. Kumar, and S. Levine. The ingredients of real-world robotic reinforcement learning. *arXiv preprint arXiv:2004.12570*, 2020.

[21] L. P. Kaelbling. Learning to achieve goals. In *IJCAI*, pages 1094–1099. Citeseer, 1993.

[22] T. Schaul, D. Horgan, K. Gregor, and D. Silver. Universal value function approximators. In *International conference on machine learning*, pages 1312–1320. PMLR, 2015.

[23] 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.

[24] 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.

[25] 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.

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

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

[28] A. Vahdat and J. Kautz. Nvae: A deep hierarchical variational autoencoder. *Advances in Neural Information Processing Systems*, 33:19667–19679, 2020.

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

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

[31] 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.

[32] S. Karaman, M. R. Walter, A. Perez, E. Frazzoli, and S. Teller. Anytime motion planning using the rrt. In *2011 IEEE International Conference on Robotics and Automation*, pages 1478–1483. IEEE, 2011.

[33] A. Vahdat and J. Kautz. Nvae: A deep hierarchical variational autoencoder. *Advances in Neural Information Processing Systems*, 33:19667–19679, 2020.

[34] S. Parisi, V. Dean, D. Pathak, and A. Gupta. Interesting object, curious agent: Learning task-agnostic exploration. *Advances in Neural Information Processing Systems*, 34:20516–20530, 2021.

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

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

[37] F. Ebert, Y. Yang, K. Schmeckpeper, B. Bucher, G. Georgakis, K. Daniilidis, C. Finn, and S. Levine. Bridge data: Boosting generalization of robotic skills with cross-domain datasets. *arXiv preprint arXiv:2109.13396*, 2021.

[38] 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.

[39] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *arXiv preprint arXiv:1801.01290*, 2018.