Appendix: Curious Representation Learning for Embodied Intelligence

Yilun Du
MIT

Chuang Gan
MIT-IBM Watson AI Lab

Phillip Isola
MIT

We analyze our CRL in natural biological setting in Section A. We provide hyper-parameter details for each setting reported in the paper in Section B, including pseudocode in Section B.6. We then provide additional quantitative numbers of Section C, quantifying diversity of gathered images in Section C.1, individual reinforcement learning runs in Section C.2, and different imitation learning runs on the val-seen subset in Section C.3.

A. Biological Exploration

Our interactive representation learning approach in Section **??** still departs from real biological learning in several important ways. While real biological learning occurs in a single environment, in Section **??**, we assume several different environments run in parallel. Furthermore, while real biological exploration occurs contiguously and persistently across time, in Section **??**, we assume each environment has a maximum duration of exploration, after which the agent teleports to a new house environment. While both assumptions are standard for RL training, in this section, we investigate how CRL and other approaches in Section **??** behave in a realistic biological setting.

We train agents using a single house from the Habitat Matterport3D dataset, using a single environment process, and assume an infinitely long episode duration. We plot the number of tiles explored in that single environment in Figure 1 and and plot of contrastive loss on gathered images in Figure 2. Overall, we find that CRL gets a comparable number of tiles explored as the learned counts-based agent, but finds much more diverse images (indicated by a much higher contrastive loss in Figure 2). When representations are evaluated for downstream real image recognition, we find the weights obtained from CRL obtain a top 5 accuracy of 11.02 compared to 9.53 from random exploration and 8.98 from learned counts exploration.

B. Training Details

We provide detailed hyper-parameters for each of the experiments in the paper. For RL agents, we utilize the PPO implementation included with Habitat baselines [3].

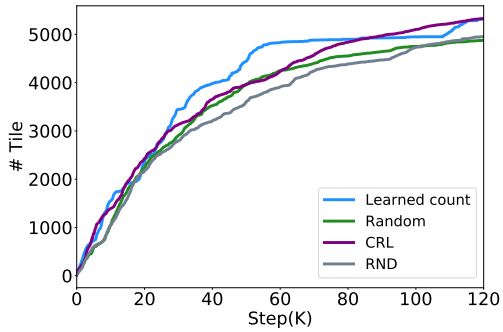

Figure 1: Plots of the average number of tiles explored in the biological setting where an agent is put in a single house environment. CRL explores comparably to a learned counts-based method.

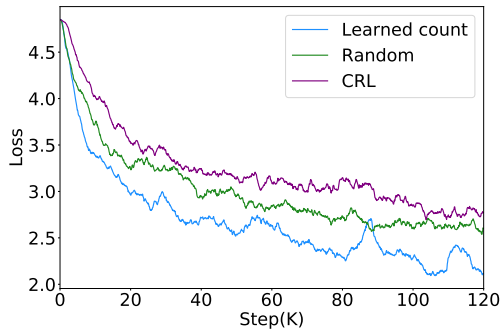

Figure 2: Plots of contrastive loss over time using different exploration methods in the biological learning setting of a single house environment. By treating the process of image gathering as an adversarial process, CRL enables the procurement of diverse images, leading to larger contrastive loss.

B.1. Pretraining

To pretrain representations, we utilize a total of 10M frames on the Habitat Matterport3D dataset. We train RL policies with 16 environments in parallel using PPO. Each individual episode has a maximum length of 500 steps. RL policies are trained with Adam with a learning rate of 0.0025, with generalized advantage estimation, an entropy coefficient of 0.01, discount factor 0.99, τ of 0.95, clip rate 0.2, with a data buffer size of 128 steps. Our RL policy is a LSTM network, with a single recurrent layer with hidden dimension of 512. Policies are updated for 4 epochs on images stored

in the data buffer.

To train our representation learning model, we update the model at the same time as the RL policy, using the stored images in the data buffer. Models are tried with Adam with a learning rate of 0.0001. For contrastive learning models, we follow SimCLR [2] and utilize a temperature of 0.07, and the default ImageNet color augmentation, crop size, and horizontal flip augmentations.

B.2. ImageNav

For the ImageNav task in the Habitat Gibson dataset, we train RL policies with 6 environments in parallel using PPO for 10 million frames. We utilize a ResNet50 to embed image goal observations (initialized with pretrained representations). Our RL policy is a recurrent network, with a single recurrent layer with hidden dimension of 512. Our RL policy is trained with PPO using Adam with a learning rate of 0.0025, a clip rate of 0.2, an entropy coefficient of 0.01, using generalized advantage estimation, with a discount factor of 0.99, τ of 0.95, and with a data buffer size of 64 steps. Policies are updated for 2 epochs on images stored in the data buffer.

B.3. ObjectNav

For the ObjectNav task in the Habitat Matterport3D dataset, we train RL policies with 16 environments in parallel using PPO for 10 million frames. We utilize a ResNet50 to embed image goal observations (initialized with pretrained representations). Our RL policy is a recurrent network, with a single recurrent layer with hidden dimension of size 512. Our RL policy is trained with PPO using Adam with a learning rate of 0.0025, a clip rate of 0.2, an entropy coefficient of 0.01, using generalized advantage estimation, with a discount factor of 0.99, τ of 0.95, and with a data buffer size of 64 steps. Policies are updated for 4 epochs on images stored in the data buffer.

B.4. Language Imitation

To train language imitation agents with either behavioral cloning or DAGGER, we directly utilize the authors' originally released repo, replacing convolutional encoders with or pretrained weights. We use Habitat-version 0.1.6 as our simulator for imitation learning.

B.5. Places Images

To finetune linear classifiers over ResNet50 average pooled features, we use the Adam optimizer with learning rate 0.001. We utilize early stopping to determine the number of training epochs to train linear classifiers, and train classifiers until the classification loss on the validation dataset increased (evaluated at the end of each training epoch).

We select the following classes in Places to apply classification over: abbey, alley, amphitheater, amusement_park, aqueduct, arch, apartment_building_outdoor, badlands, bamboo_forest, baseball_field, basilica, bayou, boardwalk, boat_deck, botanical_garden, bridge, building_facade, butte, campsite, canyon, castle, cemetery, chalet, coast, construction_site, corn_field, cottage_garden, courthouse, courtyard, creek, crevasse, crosswalk, cathedral_outdoor, church_outdoor, dam, dock, driveway, desert_sand, desert_vegetation, doorway_outdoor, excavation, fairway, fire_escape, fire_station, forest_path, forest_road, formal_garden, fountain, field_cultivated, field_wild, garbage_dump, gas_station, golf_course, harbor, herb_garden, highway, hospital, hot_spring, hotel_outdoor, iceberg, igloo, islet, ice_skating_rink_outdoor, inn_outdoor, kasbah, lighthouse, mansion, marsh, mausoleum, medina, motel, mountain, mountain_snowy, market_outdoor, monastery_outdoor, ocean, office_building, orchard, pagoda, palace, parking_lot, pasture, patio, pavilion, phone_booth, picnic_area, playground, plaza, pond, racecourse, raft, railroad_track, rainforest, residential_neighborhood, restaurant_patio, rice_paddy, river, rock_arch, rope_bridge, ruin, runway, sandbar, schoolhouse, sea_cliff, shed, shopfront, ski_resort, ski_slope, sky, skyscraper, slum, snowfield, swamp, stadium_baseball, stadium_football, swimming_pool_outdoor, television_studio, topiary_garden, tower, train_railway, tree_farm, trench, temple_east_asia, temple_south_asia, track_outdoor, underwater_coral_reef, valley, vegetable_garden, veranda, viaduct, volcano, waiting_room, water_tower, watering_hole, wheat_field, wind_farm, windmill, yard.

B.6. Pseudocode

For clarity, we present pseudocode describing the representation pretraining process of CRL in Algorithm 1

Algorithm 1 CRL pretraining algorithm.

Input: Environment E, Representation learning model M_ϕ, Policy π_θ, Buffer B
▷ *Train CRL Model:*
while not converged **do**
  ▷ *Gather information from the environment:*
  for sample K steps **do**
    $\boldsymbol{x} \leftarrow \text{get\_obs}(E)$
    $B = B \cup \boldsymbol{x}$
    $a \leftarrow \pi_\theta(\boldsymbol{x})$
    $\text{step}(E, a)$
  end for
  ▷ *Update M_ϕ, π_θ using gathered data:*
  $\mathcal{L}_\phi = \mathcal{L}_{\text{Rep}}(M_\phi, B)$
  ▷ *Compute loss for policy π_θ using reward equal to $\mathcal{L}_\phi$:*
  $\mathcal{L}_\theta = \mathcal{L}_{\text{PPO}}(\phi_\theta, B, \mathcal{L}_\phi)$
  $\Delta\phi, \Delta\theta \leftarrow \nabla_\phi \mathcal{L}_\phi, \nabla_\theta \mathcal{L}_\theta$
  Update ϕ, θ using $\Delta\phi, \Delta\theta$, through Adam:
end while

Table 1: Comparison of performance of each pretrained representation on instruction following evaluated in seen validation rooms.

Setting	Method	SPL↑	Success↑	Goal Distance↓
Behavioral Cloning	From Scratch	0.215	0.230	8.689
	RND [1]	0.210	0.228	8.536
	ATC [4]	0.210	0.223	8.379
	CRL (ours)	**0.234**	**0.248**	**8.364**
	Imagenet	0.225	0.241	8.679
Dagger	From Scratch	0.265	0.279	7.549
	RND [1]	0.284	0.302	**7.044**
	ATC [4]	0.273	0.290	7.117
	CRL (ours)	**0.295**	**0.316**	7.441
	Imagenet	0.267	0.281	7.399

C. Additional Quantitative Results

C.1. Quantitative Measures of Exploration

We quantitatively analyze the diversity of images found in the main paper Figure 5, utilizing the average distance between LPIPS embeddings of different images following [6]. We collect 2048 across each exploration method, consisting of 128 seperate images gathered over 16 different environments. We find that using the exploration policy of CRL obtains LPIPS diversity of 0.728 (0.001), while the learned counts [5] policy obtains LPIPS diversity of 0.717 (0.001) and random exploration obtains an LPIPS diversity of 0.708 (0.001), with standard error reported in parentheses calculated across gathered trajectories. Quantitatively, CRL leads to more diverse image gathering.

C.2. Reinforcement Learning Quantitative Results

We provide a table of results across the first 3 evaluated seeds in ObjectNav and ImageNav in Table 2 and Table 3. CRL performs better than other approaches.

C.3. Imitation Learning Quantitative Results

We report imitation learning results on validation seen rooms in Table 1. Using frozen representations from CRL performs well.

References

[1] Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*, 2018. 3, 4, 5

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020. 2

[3] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A platform for embodied ai research, 2019. 1

[4] Adam Stooke, Kimin Lee, Pieter Abbeel, and Michael Laskin. Decoupling representation learning from reinforcement learning, 2020. 3, 4, 5

[5] Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, OpenAI Xi Chen, Yan Duan, John Schulman, Filip DeTurck, and Pieter Abbeel. #exploration: A study of count-based exploration for deep reinforcement learning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. 3, 4, 5

[6] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A. Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation, 2018. 3

Table 2: Comparison of embodied navigation with learned interactive representations. Policies are evaluated on the test set of ImageNav tasks and are trained for 10 million frames in each environment. We report individual results of evaluated seeds. We consider either training an RL agent from scratch, utilizing existing representation learning methods (ATC [4], RND [1] and contrastive learning) or utilizing supervised weights (PointNav Policy, ImageNet Initialization). RL agents initialized from pretrained weights have visual representations frozen, while all weights in the from scratch RL agent are trained.

Environment	Category	Method	Seed	SPL↑	Soft SPL↑	Success↑	Goal Distance↓
ImageNav	From Scratch	From Scratch	0	0.0238	0.191	0.036	4.98
			1	0.0171	0.176	0.033	4.84
			2	0.0237	0.185	0.051	4.76
			3	0.0181	0.147	0.034	4.72
			4	0.0206	0.166	0.038	4.94
	Other Representation Learning Algorithms	RND [1]	0	0.0166	0.119	0.044	5.09
			1	0.0050	0.082	0.020	5.32
			2	0.0172	0.151	0.020	5.12
			3	0.0167	0.101	0.024	5.62
			4	0.0234	0.166	0.038	4.94
		ATC [4]	0	0.0183	0.133	0.060	4.59
			1	0.0339	0.209	0.063	4.69
			2	0.0231	0.180	0.043	4.64
			3	0.0350	0.190	0.063	4.84
			4	0.0237	0.146	0.070	4.49
	Contrastive Learning	Random	0	0.0320	0.204	0.058	4.70
			1	0.0315	0.198	0.054	4.73
			2	0.0268	0.192	0.046	4.83
			3	0.0238	0.155	0.061	4.58
			4	0.0277	0.217	0.048	4.60
		Learned Counts [5]	0	0.0320	0.193	0.053	4.75
			1	0.0210	0.178	0.057	4.30
			2	0.0300	0.206	0.047	4.70
			3	0.0367	0.200	0.069	4.34
			4	0.0192	0.138	0.056	4.61
		CRL (ours)	0	0.0274	0.203	0.053	4.61
			1	0.0348	0.225	0.058	4.58
			2	0.0313	0.239	0.051	4.53
			3	0.0306	0.222	0.054	4.33
			4	0.0364	0.227	0.064	4.59
	Supervised	PointNav Policy	0	0.0212	0.143	0.051	4.61
			1	0.0249	0.192	0.048	4.63
			2	0.0302	0.227	0.044	4.74
		ImageNet Initialization	0	0.0211	0.151	0.058	4.62
			1	0.0179	0.173	0.044	4.61
			2	0.0315	0.175	0.066	4.63
			3	0.0229	0.172	0.061	4.56
			4	0.0020	0.044	0.021	4.61

Table 3: Comparison of embodied navigation with learned interactive representations. Policies are evaluated on the test set of ObjectNav tasks and are trained for 10 million frames in each environment. We report individual results of the evaluated seeds. We consider either training an RL agent from scratch, utilizing existing representation learning methods (ATC [4], RND [1] and contrastive learning) or utilizing supervised weights (PointNav Policy, ImageNet Initialization). RL agents initialized from pretrained weights have visual representations frozen, while all weights in the from scratch RL agent are trained.

Environment	Category	Method	Seed	SPL↑	Soft SPL↑	Success↑	Goal Distance↓
ObjectNav	From Scratch	From Scratch	0	0.0000	0.0475	0.000	8.06
			1	0.0032	0.0487	0.010	6.93
			2	0.0000	0.0049	0.000	7.69
			3	0.0016	0.0533	0.003	7.28
			4	0.0000	0.0292	0.000	9.73
	Other Representation Learning Algorithms	RND [1]	0	0.0000	0.0043	0.000	8.06
			1	0.0000	0.0070	0.000	7.76
			2	0.0000	0.0046	0.000	7.33
			3	0.0000	0.0533	0.003	7.28
			4	0.0000	0.0081	0.000	7.53
		ATC [4]	0	0.0080	0.0881	0.010	8.37
			1	0.0000	0.0923	0.000	7.93
			2	0.0000	0.0525	0.000	8.23
			3	0.0020	0.0232	0.003	7.64
			4	0.0000	0.0334	0.000	9.44
	Contrastive Learning	Random	0	0.0041	0.0888	0.010	7.76
			1	0.0034	0.0963	0.010	7.35
			2	0.0017	0.0380	0.003	6.59
			3	0.0060	0.0669	0.010	7.52
			4	0.0060	0.0904	0.020	7.73
		Learned Counts [5]	0	0.0068	0.1174	0.029	7.26
			1	0.0075	0.1294	0.021	6.95
			2	0.0069	0.0830	0.030	8.03
			3	0.0136	0.1244	0.040	7.47
			4	0.0048	0.0972	0.010	7.75
		CRL (ours)	0	0.0304	0.1326	0.120	6.97
			1	0.0084	0.1030	0.010	7.38
			2	0.0128	0.1300	0.030	7.50
			3	0.0099	0.0960	0.000	7.74
			4	0.0202	0.1306	0.040	7.05
	Supervised	PointNav Policy	0	0.0021	0.0872	0.003	7.47
			1	0.0064	0.0881	0.009	7.15
			2	0.0310	0.1070	0.010	7.27
		ImageNet Initialization	0	0.0144	0.0610	0.020	8.14
			1	0.0046	0.0730	0.010	7.82
			2	0.0081	0.0576	0.009	7.74
			3	0.0000	0.0503	0.000	7.66
			4	0.0047	0.0690	0.010	8.18