Value-Based Abstractions for Planning

Amy Zhang

What makes representations amenable to planning?

- Structured by reachability
- Value functions make good heuristics
- How do we get good value functions for every possible downstream planning task?

Plan2Vec: embedding local reachability

Quasimetric Reinforcement Learning (QRL): Leveraging Geometric Structure in Goal-Conditioned Problems

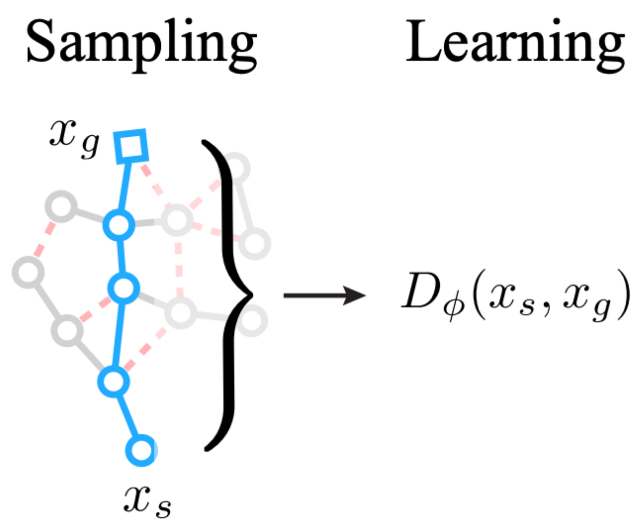
Value Implicit Pre-training (VIP): Learning Value-based Abstractions with Action-free Offline GCRL

Plan2Vec: embedding local reachability

Quasimetric Reinforcement Learning (QRL): Leveraging Geometric Structure in Goal-Conditioned Problems

Value Implicit Pre-training (VIP): Learning Value-based Abstractions with Action-free Offline GCRL

What makes representations amenable to planning?



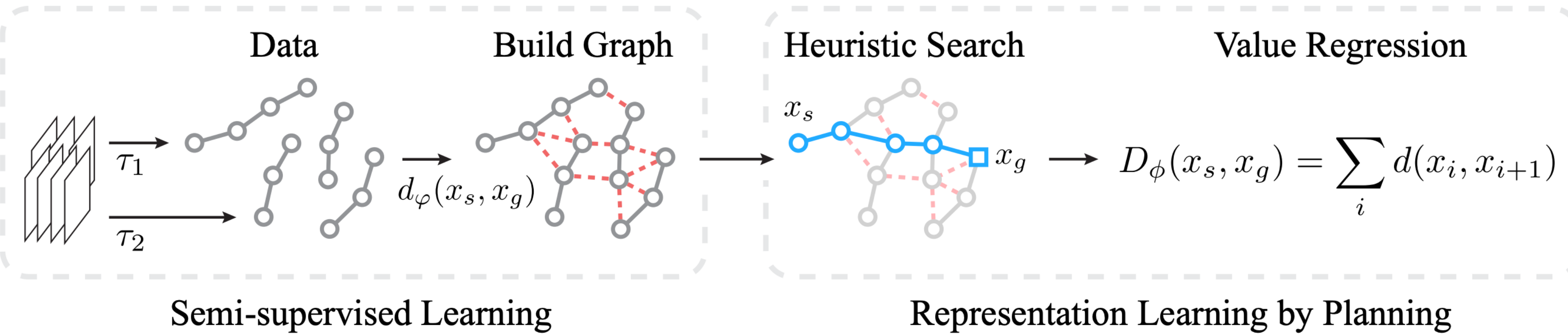
- Plan2vec is built upon the idea that for a collection of images with a local metric d, the graph G weighted by d is embedded by a Riemann manifold, the metric of which is the shortest-path-distance D.

Plan2Vec: Unsupervised Representation Learning by Latent Plans. G. Yang*, AZ*, A. Morcos, J. Pineau, P. Abbeel, R. Calandra. L4DC 2020.

Plan2Vec

Plan2vec treats the construction of the graph as a semi-supervised problem
With 3 steps:

1. Learn a local metric
2. Build a graph
3. Perform heuristic search



Plan2Vec: Unsupervised Representation Learning by Latent Plans. G. Yang*, AZ*, A. Morcos, J. Pineau, P. Abbeel, R. Calandra. L4DC 2020.

Learning a Local Metric and Building a Graph

- Temporal contrastive loss

$$L_{\text{NCE}} = -\log \frac{\exp S(x, x^+)}{\exp S(x, x^+) + \sum_i^k \exp S(x, x_i^-)}$$

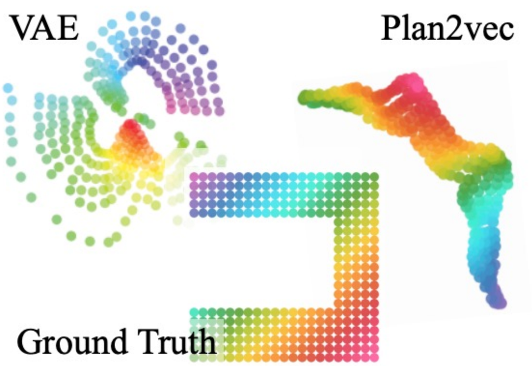
- S is *reachability*

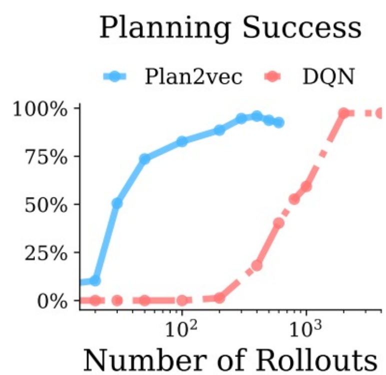
- Add edges between nodes when distance is smaller than some threshold

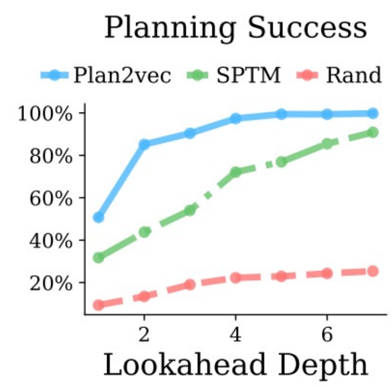
Plan2Vec: Unsupervised Representation Learning by Latent Plans. G. Yang*, AZ*, A. Morcos, J. Pineau, P. Abbeel, R. Calandra. L4DC 2020.

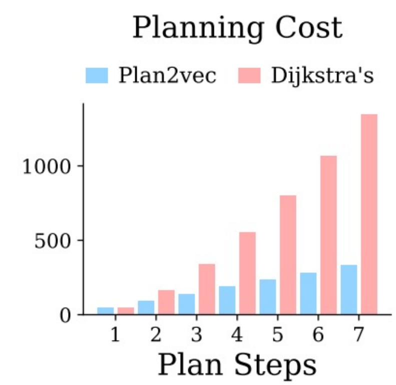
Heuristic Search

- We see global structure in learned representation (bottom left)
- Good heuristic: cheaper planning cost and higher success







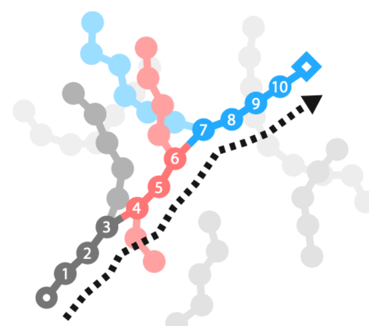


Plan2Vec: Unsupervised Representation Learning by Latent Plans. G. Yang*, AZ*, A. Morcos, J. Pineau, P. Abbeel, R. Calandra. L4DC 2020.

Key Takeaways and Insights

- Build a graph from data
- Use Dijkstra’s to construct a global metric and latent representation space.
- Learning an accurate local metric is hard!

Connecting the Dots



	Success Rate (%)		
Street Learn	Tiny	Small	Medium
Plan2vec (Ours)	$\mathbf{92.2 \pm 2.9}$	$\mathbf{57.2 \pm 4.3}$	$\mathbf{51.4 \pm 6.9}$
SPTM (1-step)	31.5 ± 5.8	19.3 ± 5.8	20.2 ± 5.2
VAE	25.5 ± 5.6	14.4 ± 4.8	16.9 ± 5.5
Random	19.9 ± 5.4	12.0 ± 5.2	12.7 ± 4.6

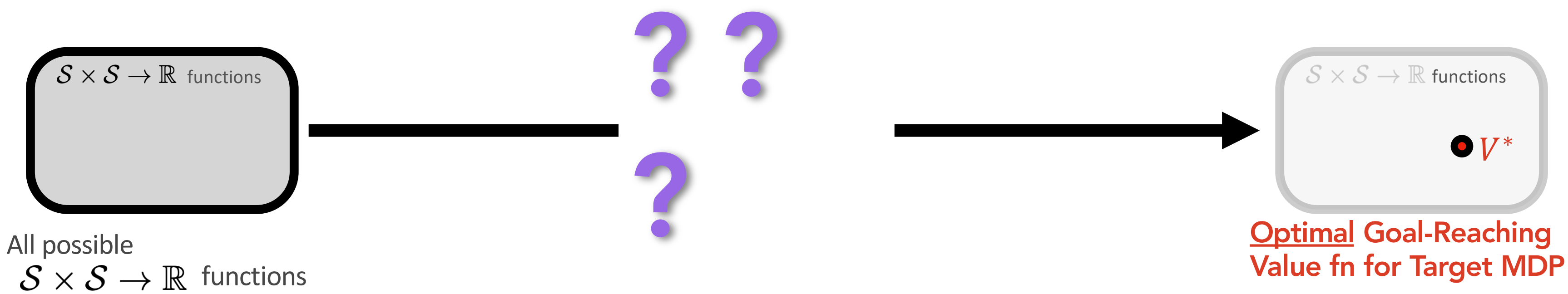
Plan2Vec: Unsupervised Representation Learning by Latent Plans. G. Yang*, AZ*, A. Morcos, J. Pineau, P. Abbeel, R. Calandra. L4DC 2020.

Plan2Vec: embedding local reachability

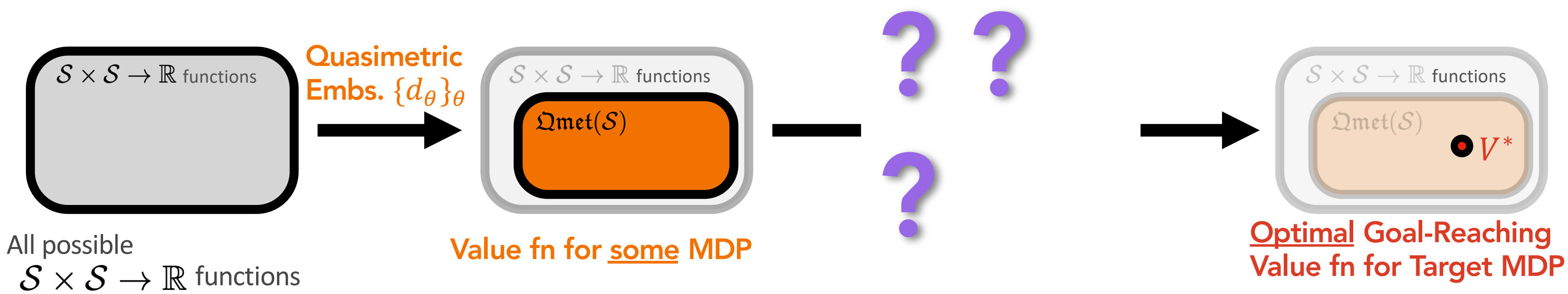
Quasimetric Reinforcement Learning (QRL): Leveraging Geometric Structure in Goal-Conditioned Problems

Value Implicit Pre-training (VIP): Learning Value-based Abstractions with Action-free Offline GCRL

Goal-Reaching Reinforcement Learning



Goal-Reaching Reinforcement Learning
+ Structures via Quasimetric Embeddings



Optimal Goal-Reaching Reinforcement Learning via Quasimetric Learning. T. Wang, A. Torralba, P. Isola, **AZ.** ICML 2023. Slide credit: Tongzhou Wang

Quasimetric RL: pull apart state-goal for global distances

Given ways to sample (e.g., from a dataset / replay buffer)

$$
\begin{aligned}
(s, a, s', \text{cost}) &\sim p_{\text{transition}} && \text{(transitions)} \\
s &\sim p_{\text{state}} && \text{(random state)} \\
s_{\text{goal}} &\sim p_{\text{goal}}, && \text{(random goal)}
\end{aligned}
$$

Quasimetric RL (QRL) optimizes a Quasimetric embedding d_θ as (negated) value function:

$$
\max_{\theta} \; \mathbb{E}_{\substack{s \sim p_{\text{state}} \\ g \sim p_{\text{goal}}}}[d_\theta(s, g)] \qquad \text{(maximize over all pairs)}
$$

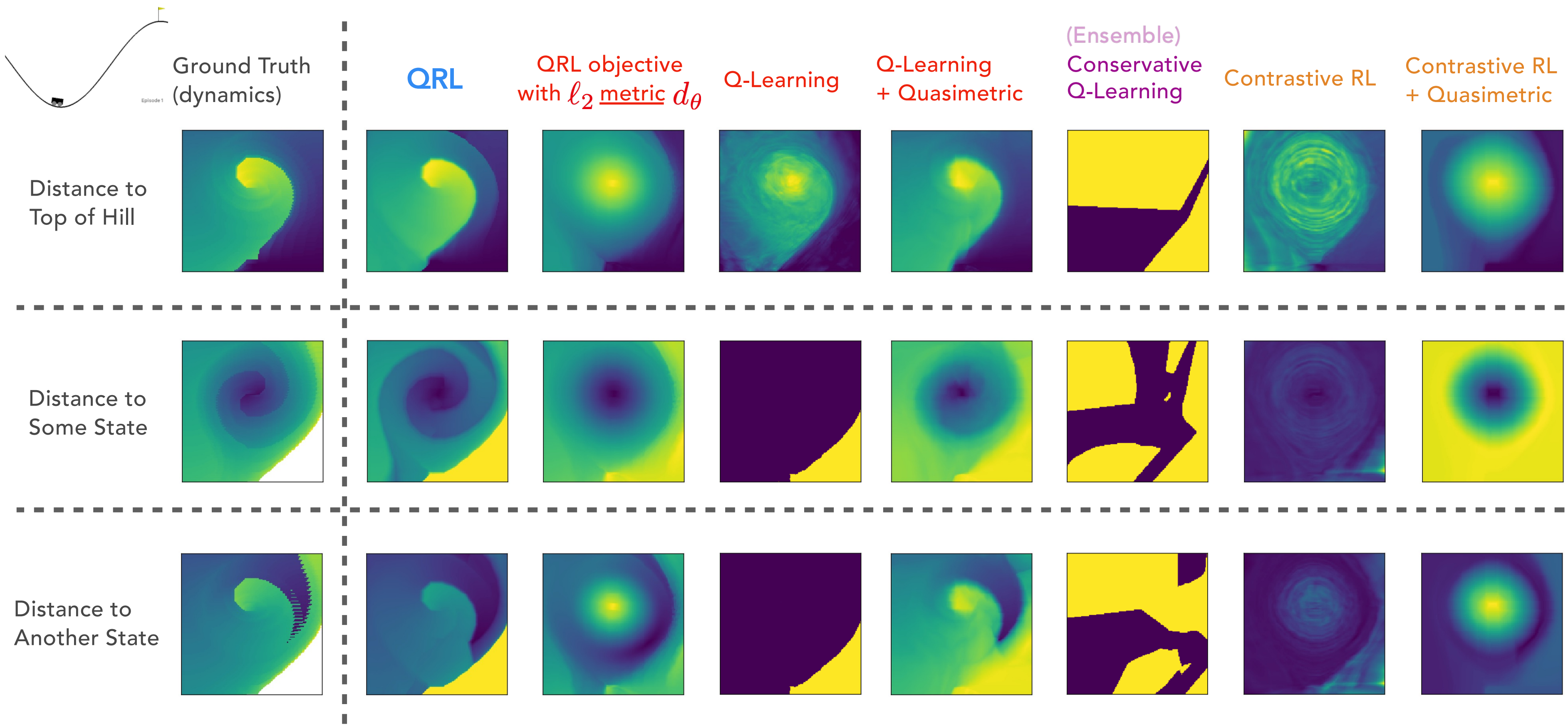
$$
\text{subject to } \mathbb{E}_{(s,a,s',\text{cost}) \sim p_{\text{transition}}}[\mathtt{relu}(d_\theta(s, s') - \text{cost})^2] \leq \underbrace{\epsilon^2}_{\epsilon > 0 \text{ small}} \qquad \text{(not overestimate local cost)}
$$

QRL Recovers Global Distances (Thm. 2&3; ICML 23)

With sufficient data and model capacity, QRL recovers optimal value fn. for any MDP.

Optimal Goal-Reaching Reinforcement Learning via Quasimetric Learning. T. Wang, A. Torralba, P. Isola, **AZ.** ICML 2023. Slide credit: Tongzhou Wang

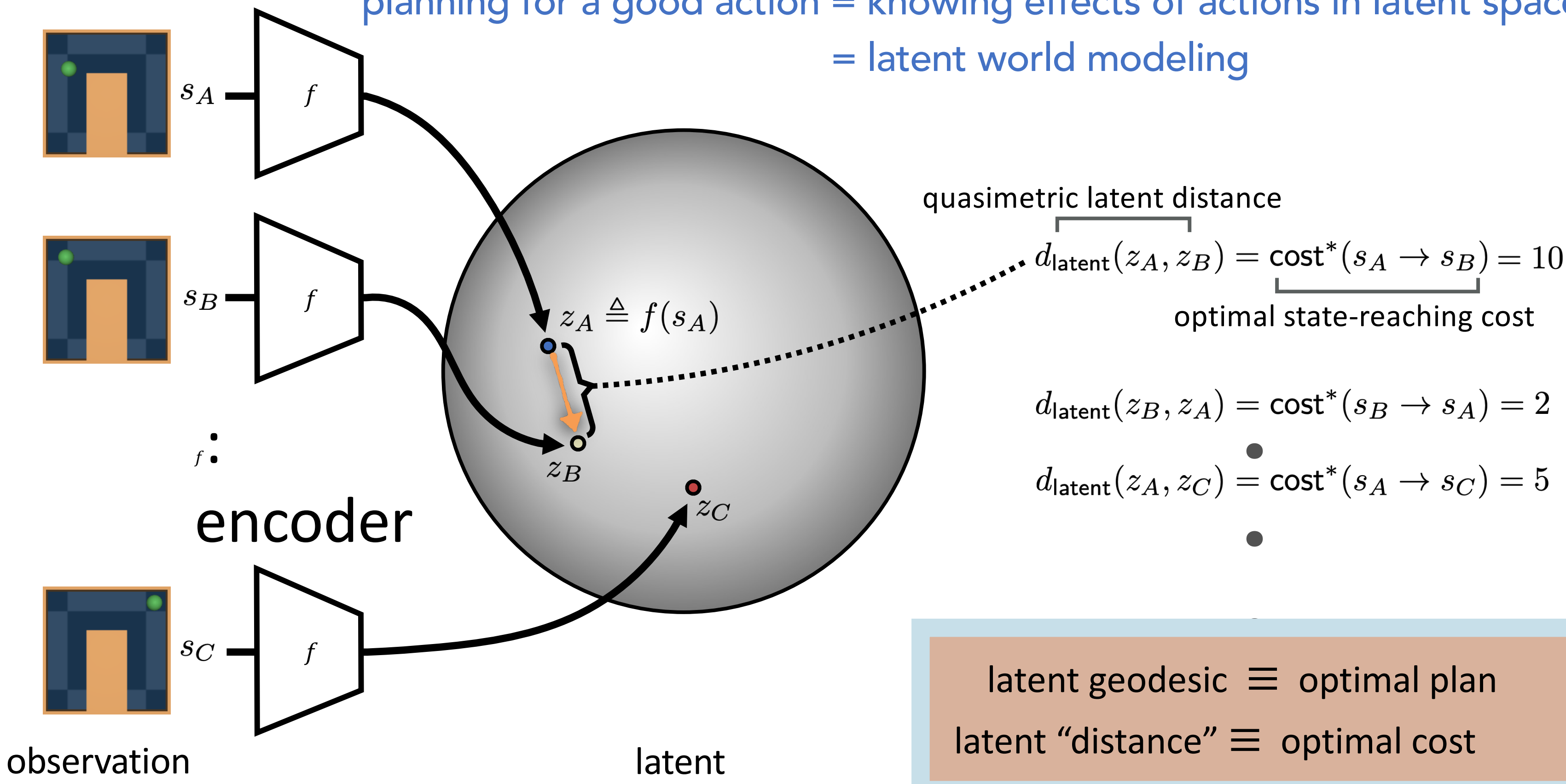
Only **QRL** recovers optimal goal-reaching value fn.



Optimal Goal-Reaching Reinforcement Learning via Quasimetric Learning. T. Wang, A. Torralba, P. Isola, **AZ.** ICML 2023. Slide credit: Tongzhou Wang

QRL learns an Optimal-Decision-Aware Representation

planning for a good action = knowing effects of actions in latent space
= latent world modeling

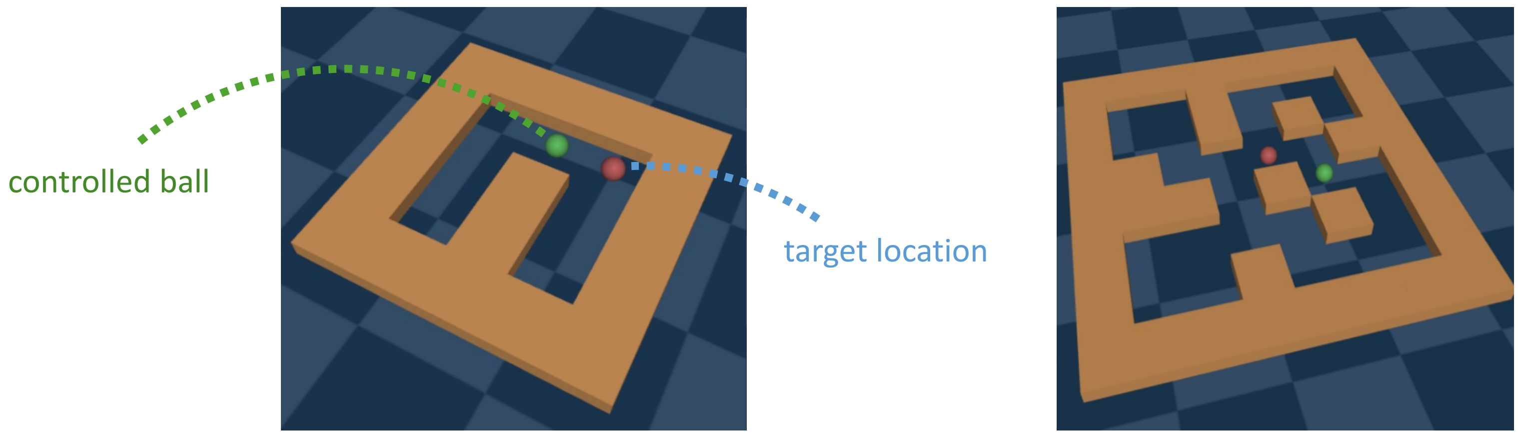


latent geodesic ≡ optimal plan
latent "distance" ≡ optimal cost

Optimal Goal-Reaching Reinforcement Learning via Quasimetric Learning. T. Wang, A. Torralba, P. Isola, **AZ.** ICML 2023. Slide credit: Tongzhou Wang

Benchmarking **QRL** (offline decision-making)

Offline RL.
Maze2D: Guide a ball through a maze toward target location.



Optimal Goal-Reaching Reinforcement Learning via Quasimetric Learning. T. Wang, A. Torralba, P. Isola, **AZ.** ICML 2023.   Slide credit: Tongzhou Wang

Benchmarking **QRL** (offline decision-making)

Offline RL.
Maze2D: Guide a ball through a maze toward target location.
QRL learns policy network by bp-ing through latent world model.

				Ensemble Q-Learning		Planning	Trajectory Modelling	
	Environment	QRL	Contrastive RL	MSG (#critic = 64)	MSG + HER (#critic = 64)	MPPI with GT Dynamics	Diffuser	Diffuser with Handcoded Controller
Single-Goal	large	**185.26** ± **28.46**	81.65 ± 43.79	159.30 ± 49.40	59.26 ± 46.70	5.1	7.98 ± 1.54	128.13 ± 2.59
	medium	**148.48** ± **46.75**	10.11 ± 0.99	57.00 ± 17.20	75.77 ± 9.02	10.2	9.48 ± 2.21	127.64 ± 1.47
	umaze	47.40 ± 23.72	95.11 ± 46.23	**101.10** ± **26.30**	55.64 ± 31.82	33.2	44.03 ± 2.25	113.91 ± 3.27
	Average	**127.05**	62.29	105.80	63.56	16.17	20.50	123.23
Multi-Goal	large	**199.19** ± **4.07**	172.64 ± 5.13	—	44.57 ± 25.30	8	13.09 ± 1.00	146.94 ± 2.50
	medium	**161.91** ± **8.10**	137.01 ± 6.26	—	99.76 ± 9.83	15.4	19.21 ± 3.56	119.97 ± 1.22
	umaze	134.11 ± 12.56	**142.43** ± **11.99**	—	27.90 ± 10.39	41.2	56.22 ± 3.90	128.53 ± 1.00
	Average	**165.07**	150.69	—	57.41	21.53	29.51	131.81

Optimal Goal-Reaching Reinforcement Learning via Quasimetric Learning. T. Wang, A. Torralba, P. Isola, **AZ.** ICML 2023. Slide credit: Tongzhou Wang

Benchmarking **QRL** (offline decision-making)

Offline RL.
Maze2D: Guide a ball through a maze toward target location.
QRL learns policy network by bp-ing through latent world model.

Ensemble Q-Learning

Planning + QRL V^*

Trajectory Modelling + QRL V^*

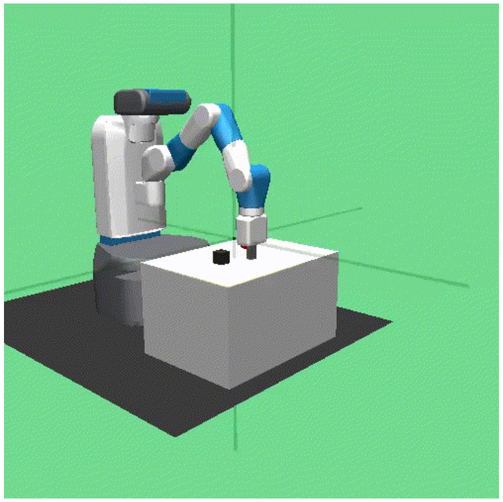
	Environment	QRL	Contrastive RL	MSG (#critic = 64)	MSG + HER (#critic = 64)	MPPI with GT Dynamics	MPPI with QRL Value	Diffuser	Diffuser with QRL Value Guidance	Diffuser with Handcoded Controller
Single-Goal	large	**185.26** ± **28.46**	81.65 ± 43.79	159.30 ± 49.40	59.26 ± 46.70	5.1	4.67 ± 5.31	7.98 ± 1.54	11.33 ± 1.48	128.13 ± 2.59
	medium	**148.48** ± **46.75**	10.11 ± 0.99	57.00 ± 17.20	75.77 ± 9.02	10.2	60.89 ± 40.38	9.48 ± 2.21	10.52 ± 3.26	127.64 ± 1.47
	umaze	47.40 ± 23.72	95.11 ± 46.23	**101.10** ± **26.30**	55.64 ± 31.82	33.2	45.88 ± 9.32	44.03 ± 2.25	42.19 ± 4.23	113.91 ± 3.27
	Average	**127.05**	62.29	105.80	63.56	16.17	37.15	20.50	21.35	123.23
Multi-Goal	large	**199.19** ± **4.07**	172.64 ± 5.13	—	44.57 ± 25.30	8	54.04 ± 7.47	13.09 ± 1.00	21.78 ± 2.86	146.94 ± 2.50
	medium	**161.91** ± **8.10**	137.01 ± 6.26	—	99.76 ± 9.83	15.4	71.24 ± 6.69	19.21 ± 3.56	33.68 ± 2.82	119.97 ± 1.22
	umaze	134.11 ± 12.56	**142.43** ± **11.99**	—	27.90 ± 10.39	41.2	84.72 ± 7.69	56.22 ± 3.90	69.49 ± 3.85	128.53 ± 1.00
	Average	**165.07**	150.69	—	57.41	21.53	70.00	29.51	41.65	131.81

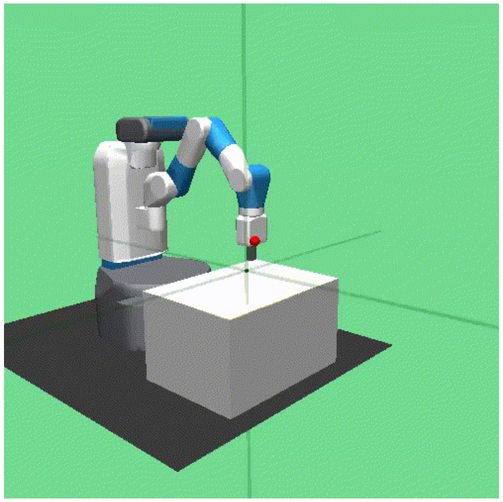
Optimal Goal-Reaching Reinforcement Learning via Quasimetric Learning. T. Wang, A. Torralba, P. Isola, **AZ.** ICML 2023. Slide credit: Tongzhou Wang

Benchmarking **QRL** (online decision-making)

Online GCRL benchmark. Control a robot to perform tasks, e.g., pushing a block.

More complex environments. Continuous actions.





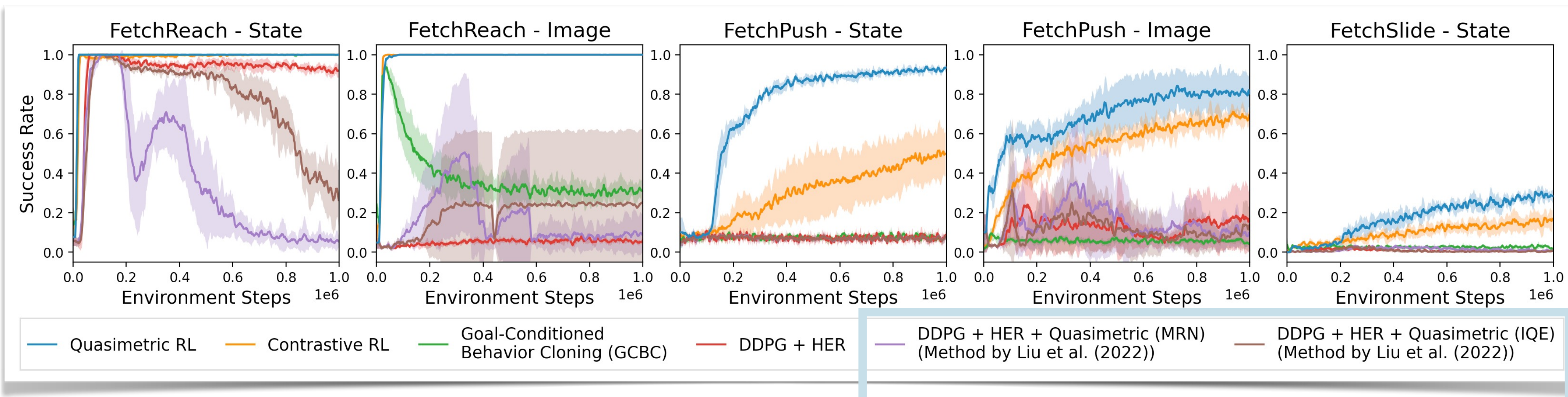
Optimal Goal-Reaching Reinforcement Learning via Quasimetric Learning. T. Wang, A. Torralba, P. Isola, **AZ.** ICML 2023. Slide credit: Tongzhou Wang

Benchmarking **QRL** (online decision-making)

Online GCRL benchmark. Control a robot to perform tasks, e.g., pushing a block.

More complex environments. Continuous actions.

QRL learns policy network by bp-ing through latent world model.



Continuous actions
Quasimetric with TD fails

Optimal Goal-Reaching Reinforcement Learning via Quasimetric Learning. T. Wang, A. Torralba, P. Isola, **AZ.** ICML 2023.

Plan2Vec: embedding local reachability

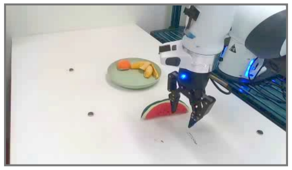
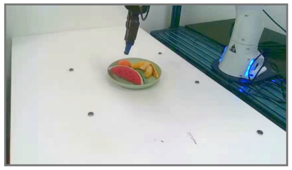
Quasimetric Reinforcement Learning (QRL): Leveraging Geometric Structure in Goal-Conditioned Problems

Value Implicit Pre-training (VIP): Learning Value-based Abstractions with Action-free Offline GCRL

Learning a Universal Value Function

Observation

Goal

V*( , )

V* = General Notion of Goal-Directed Task Progress

Rich Visual Representation so that V* can be expressed

V* itself can be used to construct visual-goal rewards for any task

VIP: Towards Universal Visual Reward and Representation via Value-Implicit Pre-Training. J.Ma, S. Sodhani, D. Jayaraman, O. Bastani, V. Kumar, **AZ**. ICLR 2023.

Slide credit: Jason Ma

Key Idea: Learning from Human Videos as a *BIG Offline Goal-Conditioned RL* Problem

Offline Dataset:
Diverse Human Videos

$$\max_{\pi_H, \phi} \mathbb{E}_{\pi^H}\left[\sum_t \gamma^t r(o; g)\right] - D_{\mathrm{KL}}(d^{\pi_H}(o, a^H; g) \| d^D(o, \tilde{a}^H; g)),$$

- Mathematically Sound
- What are human actions?
- Can't be optimized in practice

Human videos are rich sources of goal-directed behavior!

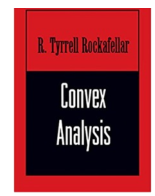
VIP: Towards Universal Visual Reward and Representation via Value-Implicit Pre-Training. J.Ma, S. Sodhani, D. Jayaraman, O. Bastani, V. Kumar, **AZ**. ICLR 2023.

Slide credit: Jason Ma

Offline *Value* Learning on Human Videos

Offline Dataset:
Diverse Human Videos

$$\max_{\pi_H,\phi} \mathbb{E}_{\pi^H}\left[\sum_t \gamma^t r(o;g)\right] - D_{\mathrm{KL}}(d^{\pi_H}(o,a^H;g)\|d^D(o,\tilde{a}^H;g)),$$



Dual Problem depends only on offline data! No dependence on actions!

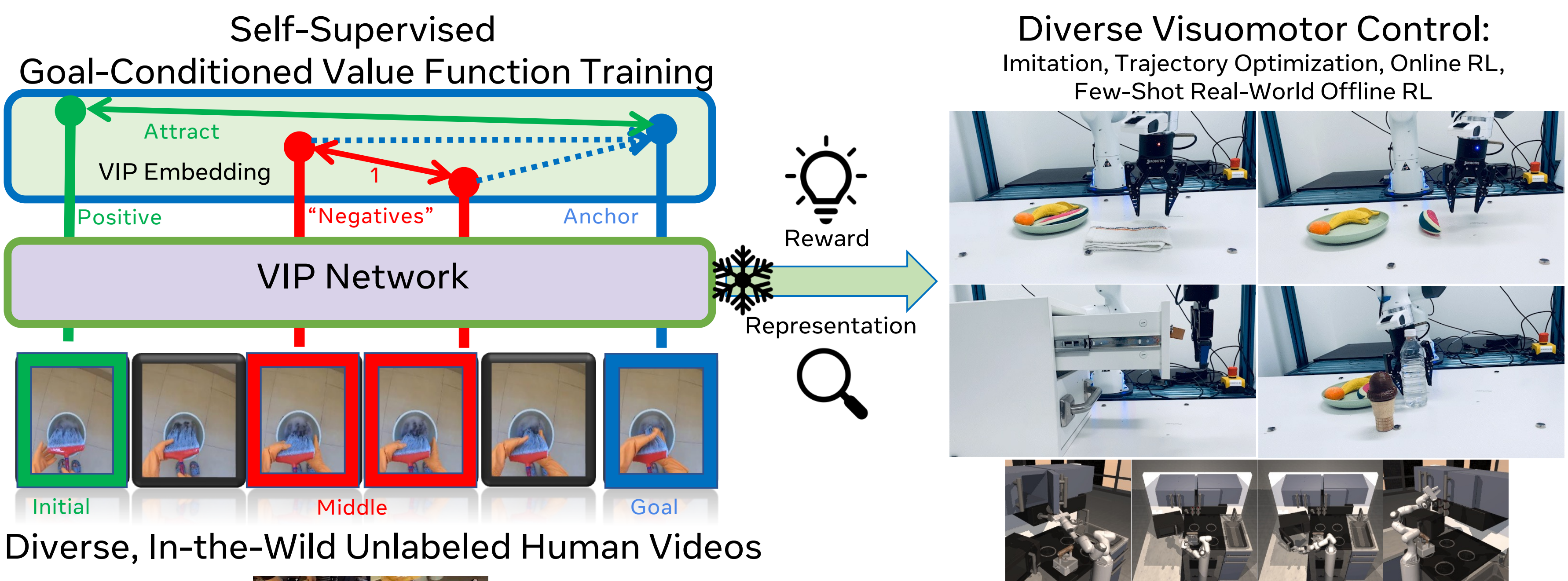
$$\max_{\phi}\min_{V} \mathbb{E}_{\underline{p(g)}}\left[(1-\gamma)\mathbb{E}_{\underline{\mu_0(o;g)}}[V(\phi(o);\phi(g))] + \log \mathbb{E}_{\underline{(o,o';g)}\sim D}\left[\exp\left(r(o,g)+\gamma V(\phi(o');\phi(g)) - V(\phi(o),\phi(g))\right)\right]\right]$$

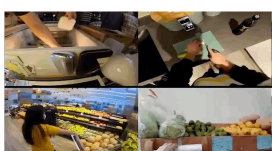
goal frame initial frame middle frame

VIP: Towards Universal Visual Reward and Representation via Value-Implicit Pre-Training. J.Ma, S. Sodhani, D. Jayaraman, O. Bastani, V. Kumar, **AZ.** ICLR 2023.

Slide credit: Jason Ma

VIP: Towards Universal Visual Reward and Representation Via Value-Implicit Pre-Training



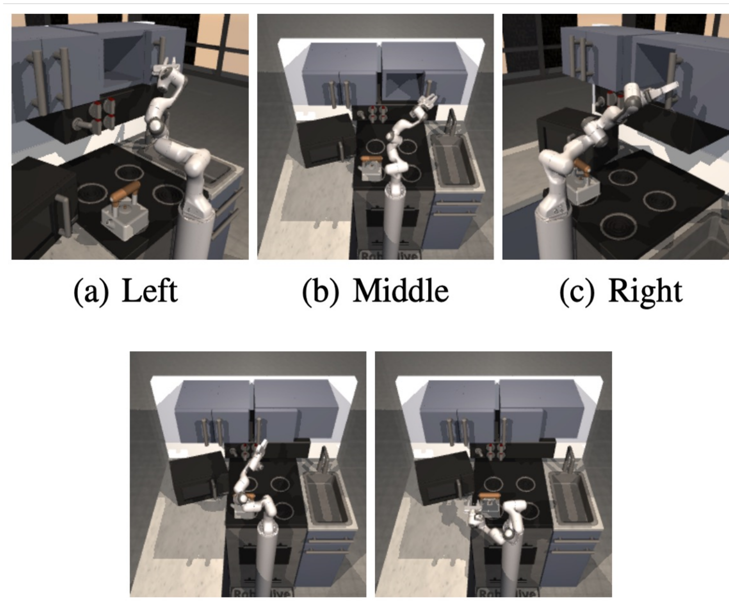


VIP: Towards Universal Visual Reward and Representation via Value-Implicit Pre-Training. J.Ma, S. Sodhani, D. Jayaraman, O. Bastani, V. Kumar, **AZ**. ICLR 2023.

Slide credit: Jason Ma

Task Variation: 3 viewpoints, 2 initial distributions

- 12 FrankaKitchen tasks covering wide-range of manipulation skills
- 3 camera views for each task
- 2 initial states (Easy, Hard)



(a) Left (b) Middle (c) Right

(d) Easy (e) Hard

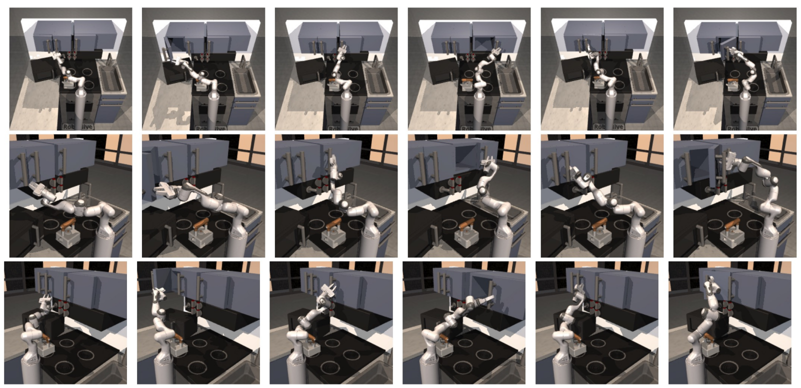


Figure 3: Frankakitchen example goal images.

Slide credit: Jason Ma

Trajectory Optimization

- Use MPPI to optimize a trajectory
  - Use the simulator to rollout proposed action sequences
  - Use pre-trained rewards to evaluate rollouts and take the first action of the best sequence
  - Repeat
- Evaluating representations’ capability as pure visual rewards
  - no policy learning (yet)

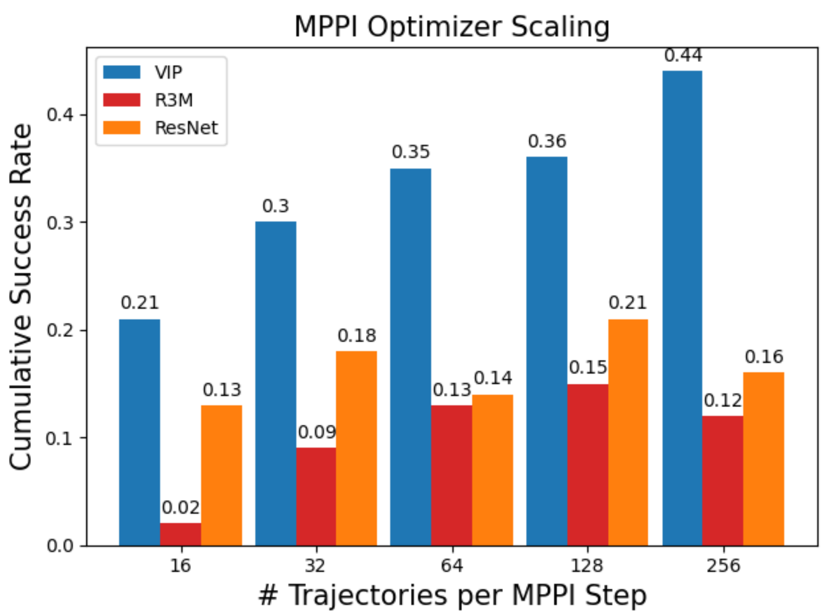
Slide credit: Jason Ma

Trajectory Optimization Result

✔ VIP robustly minimizes both robot and object pose errors!

Slide credit: Jason Ma

Scaling to Optimization Budget



Baselines exploit their unsmooth reward landscapes

VIP benefits from increasing optimization computes

Slide credit: Jason Ma

VIP Reward Weighted Regression (RWR)

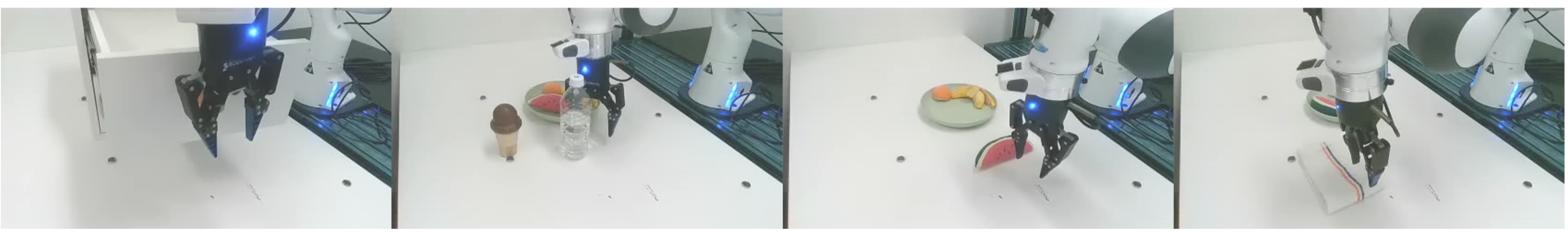
$$\mathcal{L}(\pi) = -\mathbb{E}_{D_{\text{task}}(o,a,o',g)} \left[\exp(\tau \cdot R(o, o'; \phi, g)) \log \pi(a \mid \phi(o)) \right],$$

- Weighs transitions according to pre-trained rewards
- Able to pay attention to key frames if the reward is good
- One line change from BC
- Hypothesis: VIP-RWR > VIP-BC (BC on the VIP representation)

Slide credit: Jason Ma

Tasks and Demonstrations

Environment	Object Type	Dataset	Success Criterion
CloseDrawer	Articulated Object	10 demos + 20 failures	the drawer is closed enough that the spring loads.
PushBottle	Transparent Object	20 demonstrations	the bottle is parallel to the goal line set by the icecream cone.
PlaceMelon	Soft Object	20 demonstrations	the watermelon toy is fully placed in the plate.
FoldTowel	Deformable Object	20 demonstrations	the bottom half of the towel is cleanly covered by the top half.



Slide credit: Jason Ma

Results

Table 1: Real-robot offline RL results (success rate % averaged over 10 rollouts with standard deviation reported).

Environment	*Pre-Trained* VIP-RWR	VIP-BC	R3M-RWR	R3M-BC	*In-Domain* Scratch-BC	VIP-RWR	VIP-BC
CloseDrawer	**100** $\pm$ 0	50 $\pm$ 50	80 $\pm$ 40	10 $\pm$ 30	30 $\pm$ 46	0 $\pm$ 0	0^* $\pm$ 0
PushBottle	**90** $\pm$ 30	50 $\pm$ 50	70 $\pm$ 46	50 $\pm$ 50	40$\pm$ 48	0^* $\pm$ 0	0^* $\pm$ 0
PlaceMelon	**60** $\pm$ 48	10 $\pm$ 30	0 $\pm$ 0	0 $\pm$ 0	0 $\pm$ 0	0^* $\pm$ 0	0^* $\pm$ 0
FoldTowel	**90** $\pm$ 30	20 $\pm$ 40	0 $\pm$ 0	0 $\pm$ 0	0 $\pm$ 0	0^* $\pm$ 0	0^* $\pm$ 0

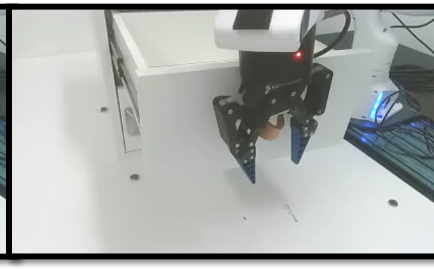
✓ Pre-training is necessary for few-shot ORL, and VIP is uniquely effective for it

Slide credit: Jason Ma

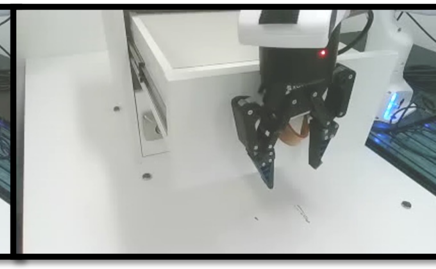
CloseDrawer & PushBottle

VIP-RWR (100%)

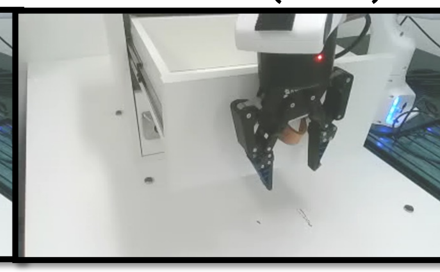
VIP-BC (50%)



R3M-RWR (90%)



R3M-BC (10%)



VIP-RWR (90%)

VIP-BC (50%)

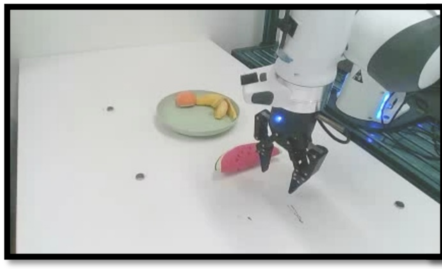
R3M-RWR (70%)

R3M-BC (50%)

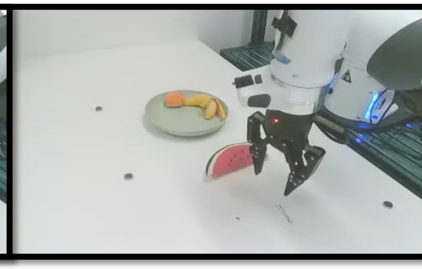
Slide credit: Jason Ma

PickPlaceMelon & FoldTowel

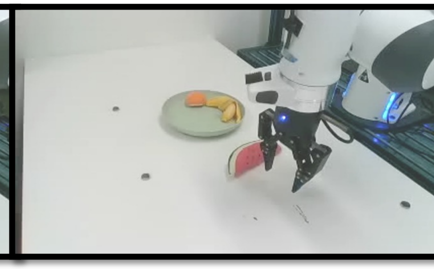
VIP-RWR (100%)



VIP-BC (50%)

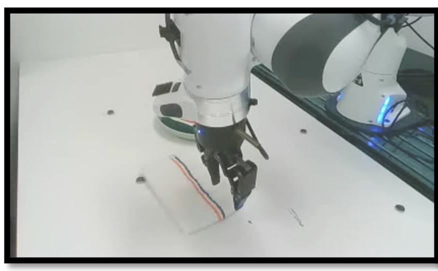


R3M-RWR (90%)



R3M-BC (10%)

VIP-RWR (90%)



VIP-BC (50%)

R3M-RWR (70%)

R3M-BC (50%)

Slide credit: Jason Ma

Open Questions

- What properties do we want in a latent representation for planning?
  - What information is needed?
  - What type of structural properties are good?
- What problems are most suited to planning?
  - All problems, or only a subset?
- Can we define a purely local learning objective that leads to global optimality? (Beyond bootstrapping)