NEURAL INFORMATION PROCESSING SYSTEMS

Microsoft Research 微软亚洲研究院

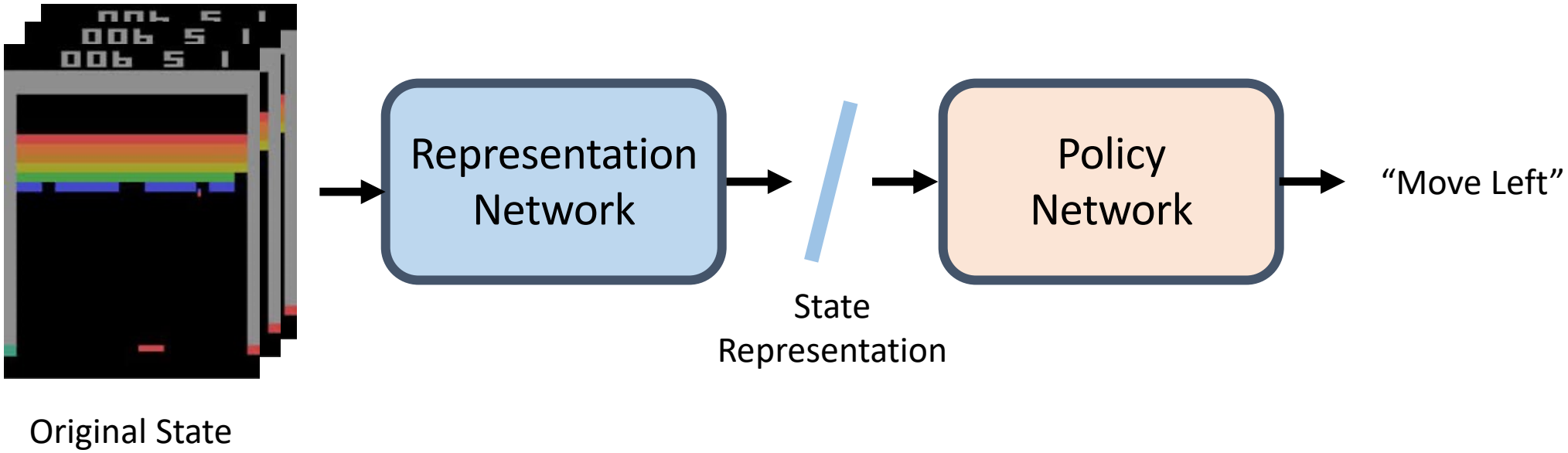
Mask-based Latent Reconstruction for Reinforcement Learning

- ***Tao Yu***[1], *Zhizheng Zhang*[2], *Cuiling Lan*[2], *Yan Lu*[2], *Zhibo Chen*[1]
- [1]*University of Science and Technology of China,* [2]*Microsoft Research Asia*

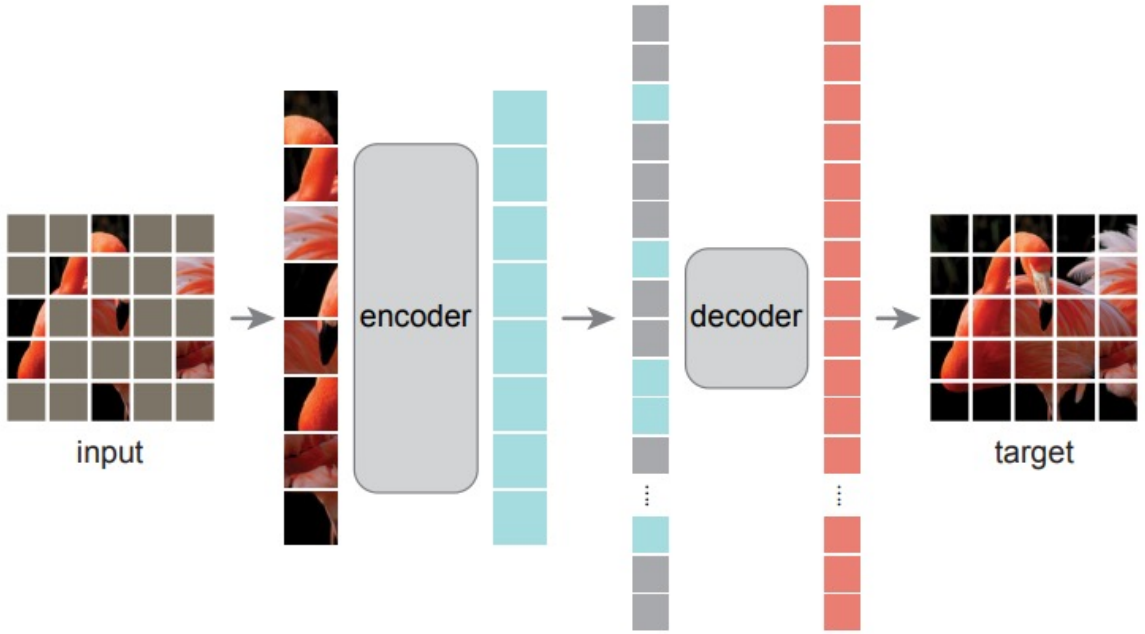
NeurIPS 2022

Scan Me for Code

Representations in RL



Masked Image Modeling



Masked Autoencoders (MAE) [1]

“What I cannot create, I do not understand.” — Richard Feynman

[1] *He, Kaiming, et al. "Masked autoencoders are scalable vision learners." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022.*

Motivation of MLR

- **Does mask-based modeling work in RL?**
- **How should we adopt the idea?**

Challenge

C1: Unlike pretraining on static datasets, RL agents learn from interactions with environments.

Challenge

C1: Unlike pretraining on static datasets, RL agents learn from interactions with environments.

C2: Visual signals with high information density usually contain distractions and redundancies for policy learning.

Challenge

C1: Unlike pretraining on static datasets, RL agents learn from interactions with environments.

C2: Visual signals with high information density usually contains distractions and redundancies for policy learning.

C3: RL data is sequential while data in MAE is single-point.

Challenge

C1: Unlike pretraining on static datasets, RL agents learn from interactions with environments.

Mask-based modeling as an auxiliary objective

C2: Visual signals with high information density usually contains distractions and redundancies for policy learning.

C3: RL data is sequential while data in MAE is single-point.

Challenge

C1: Unlike pretraining on static datasets, RL agents learn from interactions with environments.

Masked modeling as an auxiliary objective

C2: Visual signals with high information density usually contains distractions and redundancies for policy learning.

Reconstruction in the ***latent*** space

C3: RL data is sequential while data in MAE is single-point.

Challenge

C1: Unlike pretraining on static datasets, RL agents learn from interactions with environments.

Masked modeling as an auxiliary objective

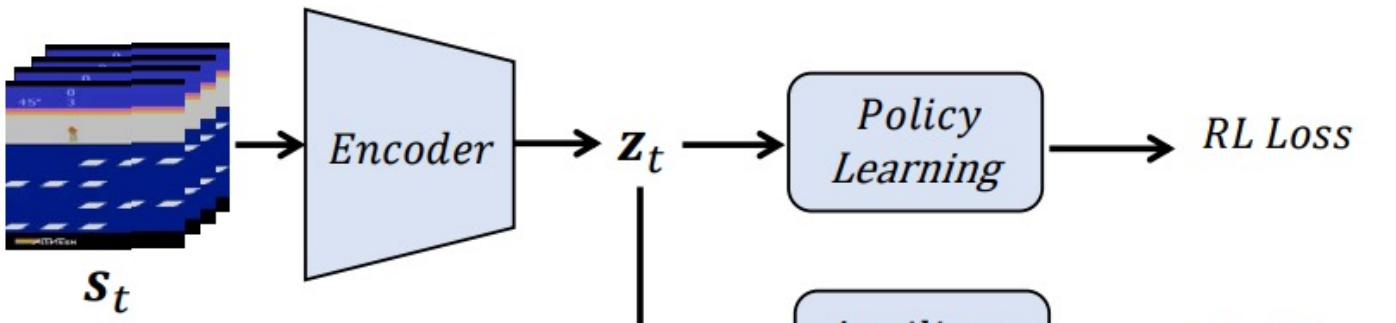
C2: Visual signals with high information density usually contains distractions and redundancies for policy learning.

Reconstruction in the ***latent*** space

C3: RL data is sequential while data in MAE is single-point.

Masked “Video” Modeling

Overall Framework

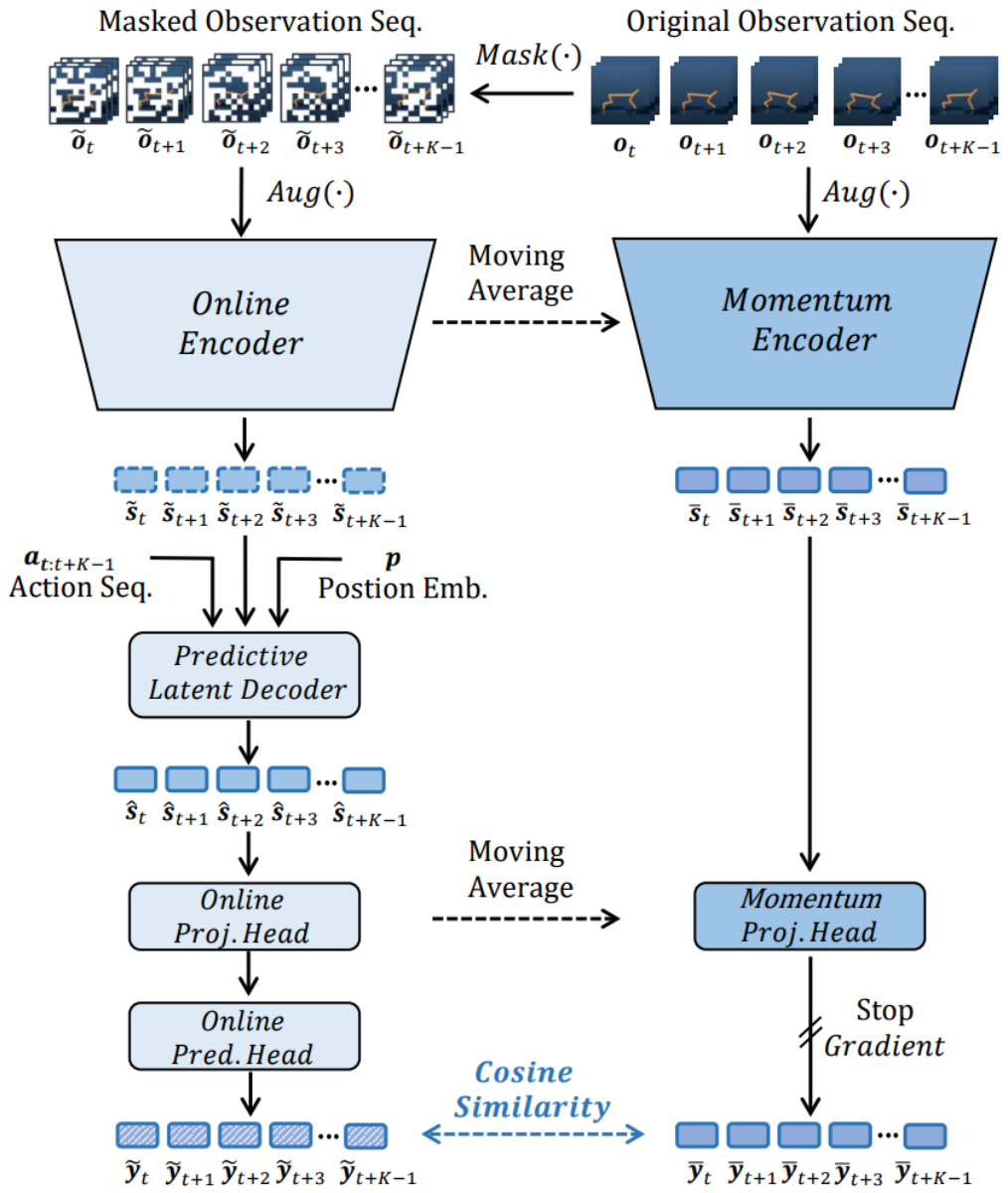


$$\mathcal{L}_{total} = \mathcal{L}_{rl} + \lambda \mathcal{L}_{mlr}$$

MLR Auxiliary Loss

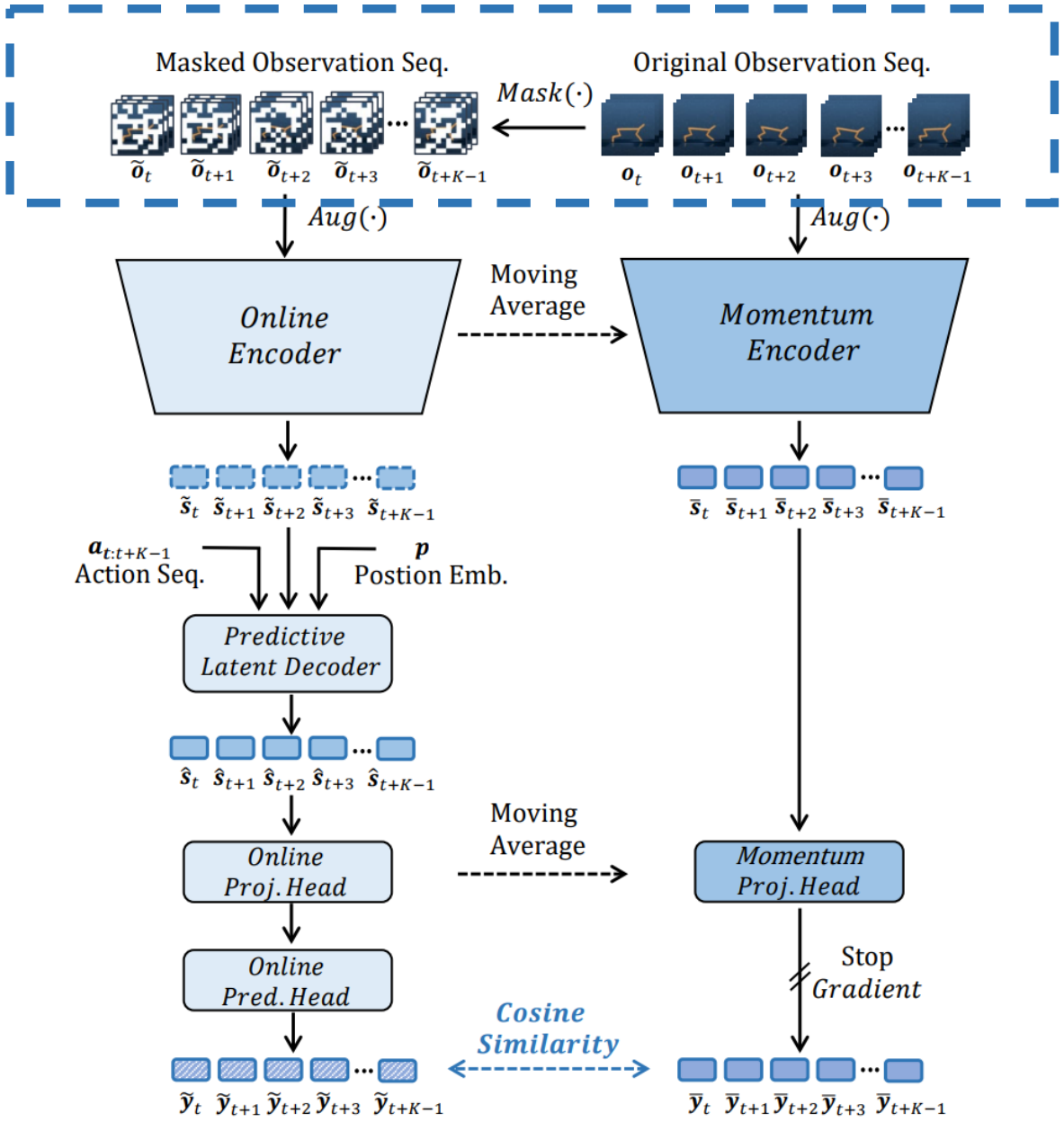
Framework

- Masking
- Encoding
- Decoding
- Reconstruction



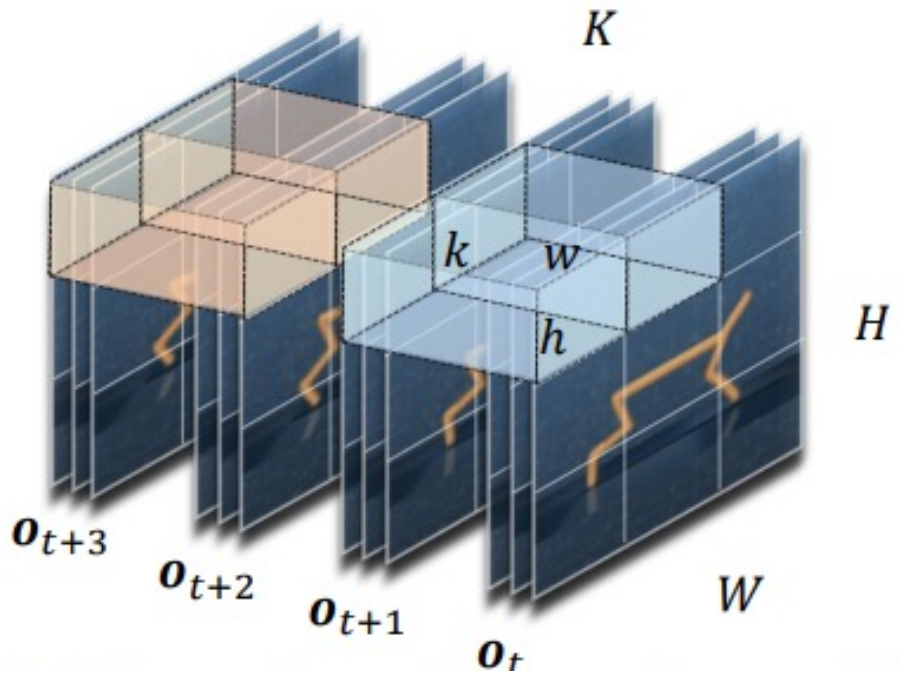
Framework

- **Masking**
- Encoding
- Decoding
- Reconstruction



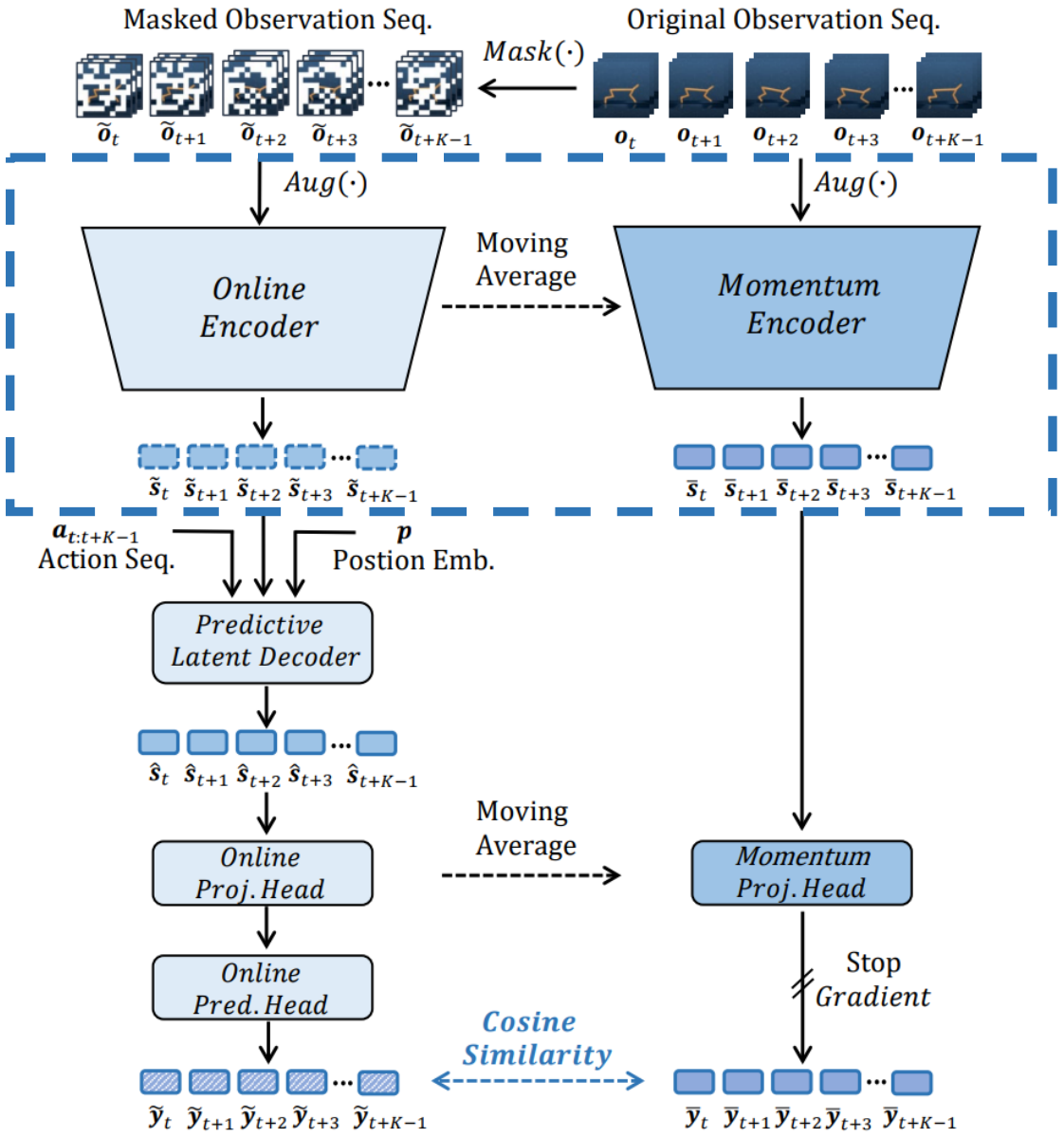
Framework

- **Masking**
- Encoding
- Decoding
- Reconstruction



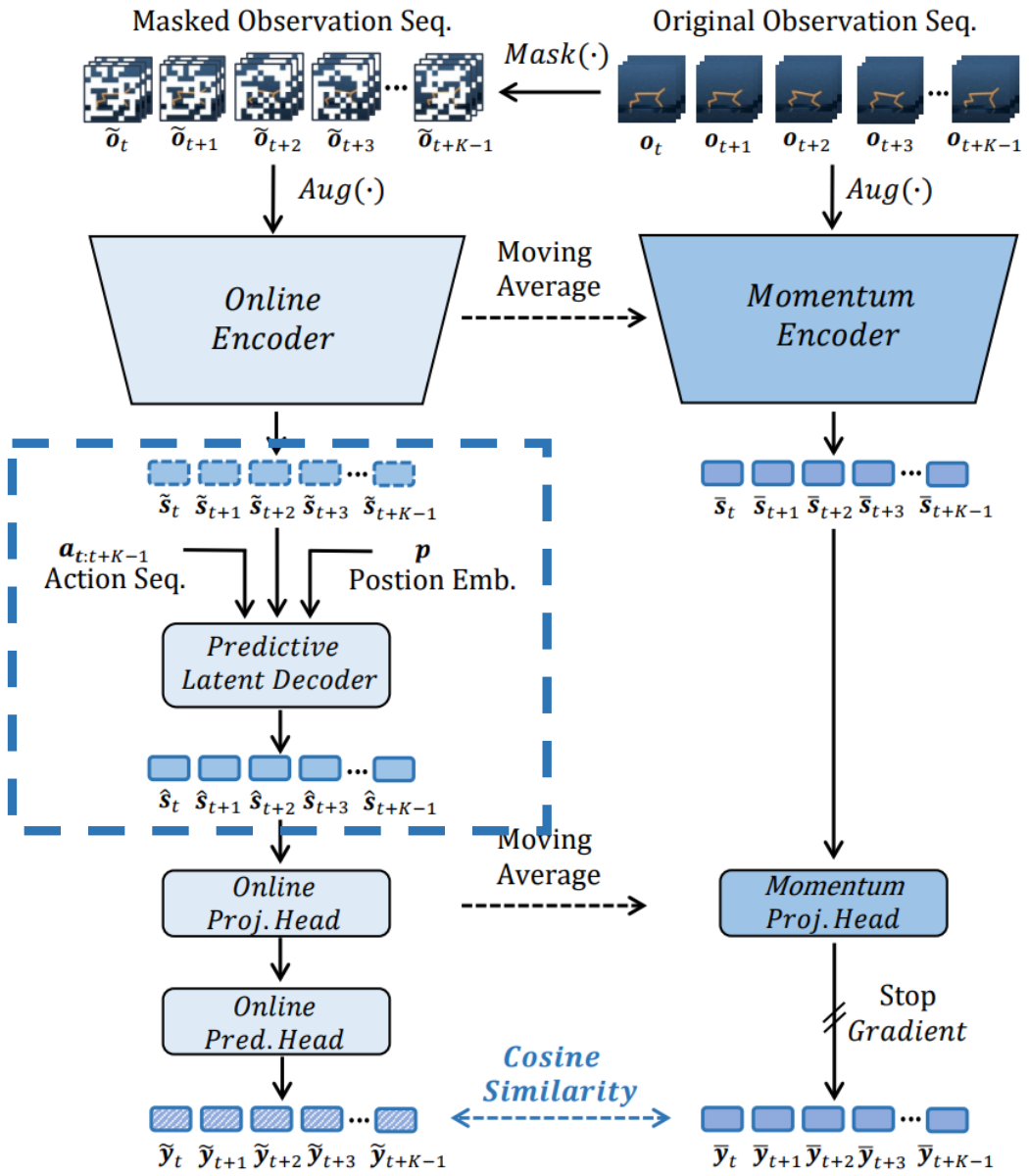
Framework

- Masking
- **Encoding**
- Decoding
- Reconstruction



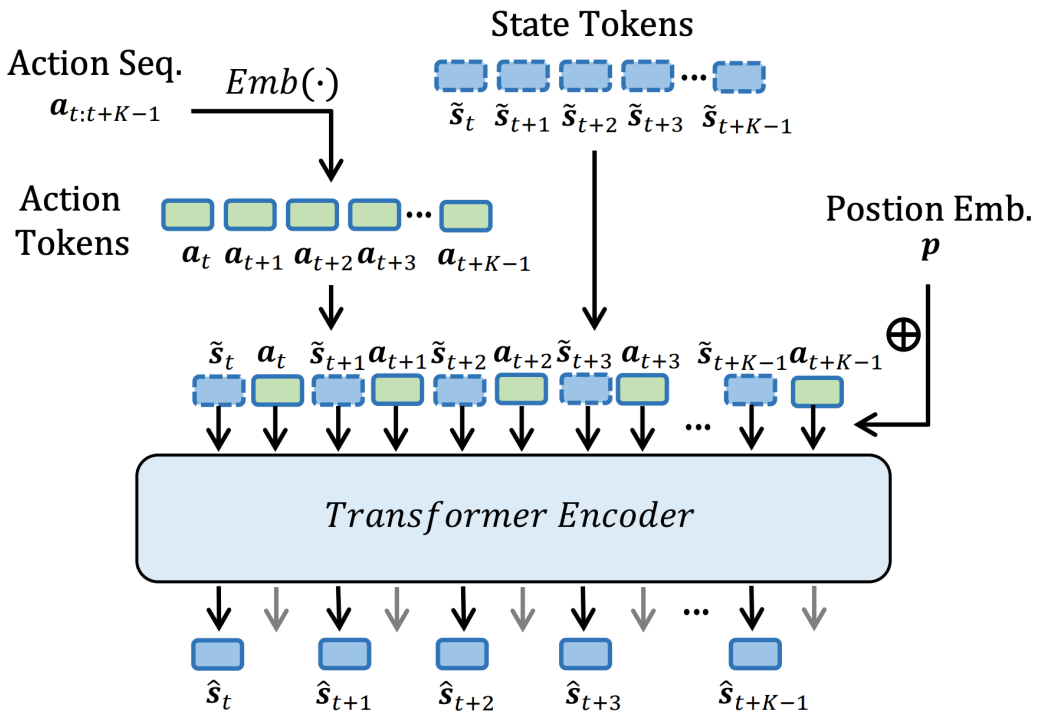
Framework

- Masking
- Encoding
- **Decoding**
- Reconstruction



Framework

- Masking
- Encoding
- **Decoding**
- Reconstruction

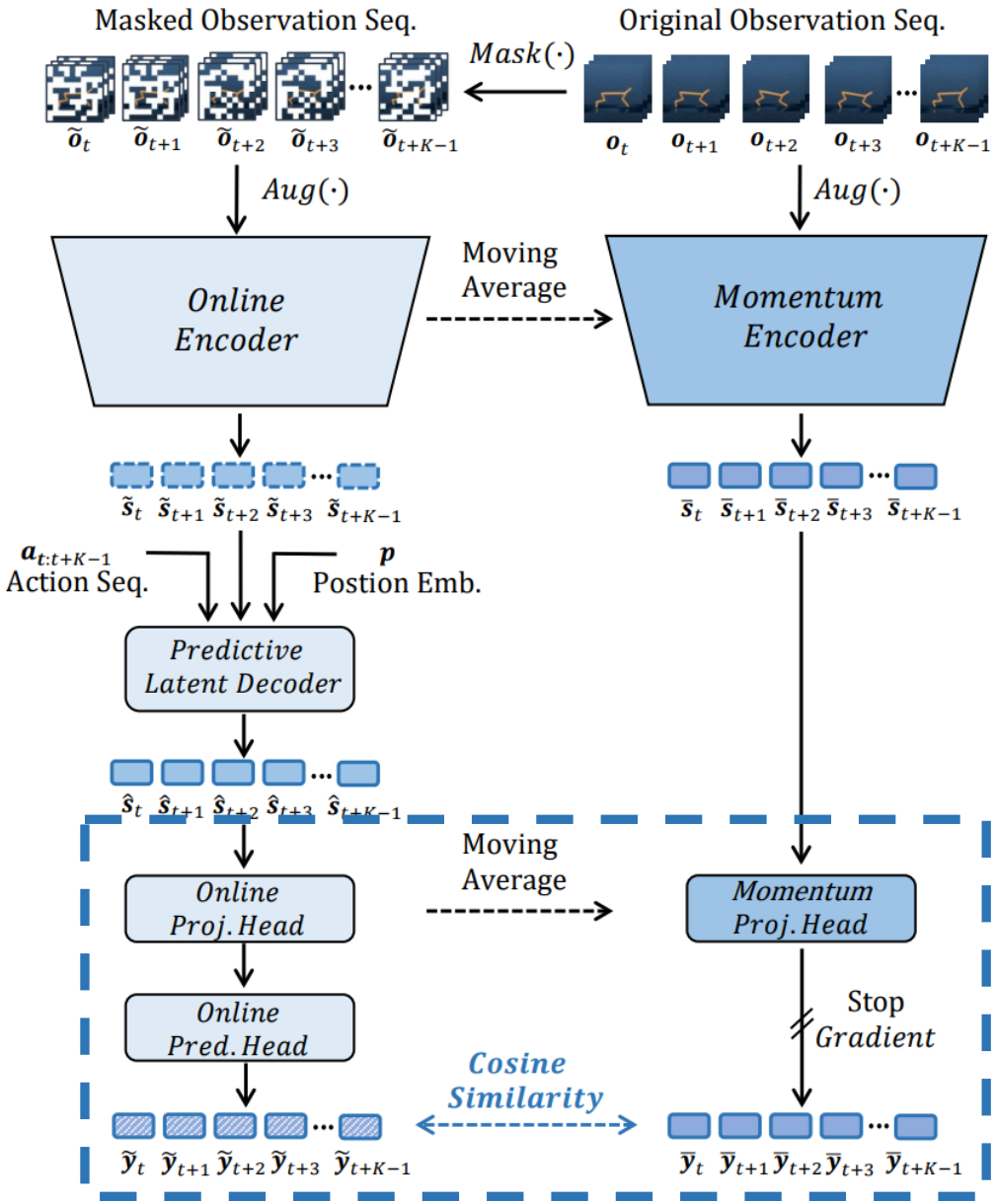


Architecture of Predictive Latent Decoder

Framework

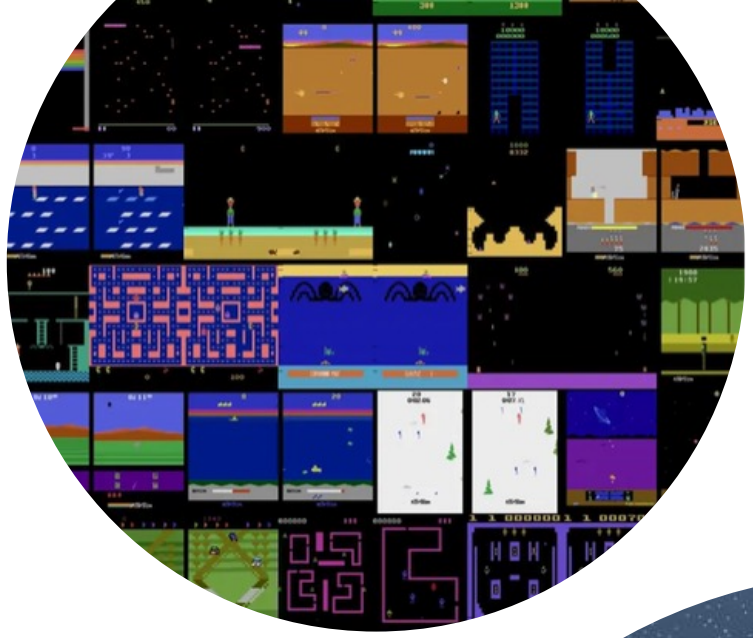
- Masking
- Encoding
- Decoding
- **Reconstruction**

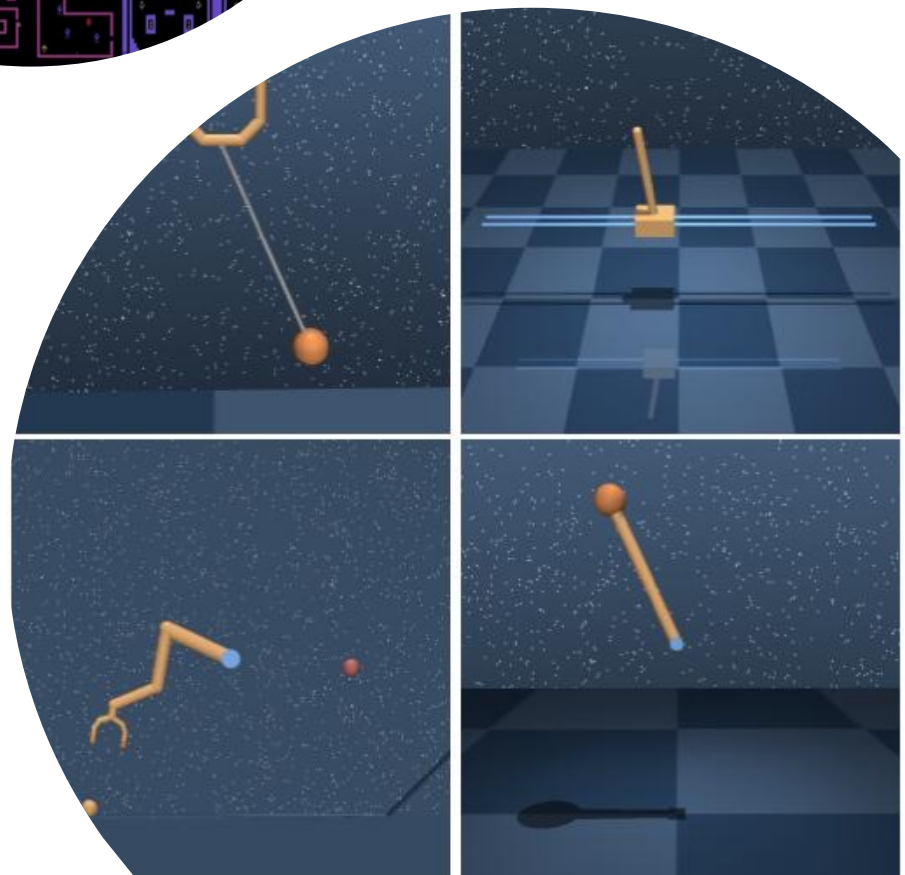
$$\mathcal{L}_{mlr} = 1 - \frac{1}{K}\sum_{i=0}^{K-1} \frac{\hat{\mathbf{y}}_{t+i}}{\|\hat{\mathbf{y}}_{t+i}\|_2} \frac{\bar{\mathbf{y}}_{t+i}}{\|\bar{\mathbf{y}}_{t+i}\|_2}$$



Experiment

- Atari-100k
- DMControl-100k, DMControl-500k





Experiment

Table 1: Comparison on the Atari-100k benchmark. Our method augments Baseline with the MLR objective and achieves a 47.9% relative improvement on IQM.

Game	Human	Random	DER	OTR	CURL	DrQ	SPR	PlayVirtual	Baseline	MLR
Alien	7127.7	227.8	802.3	570.8	711.0	734.1	841.9	947.8	678.5	**990.1**
Amidar	1719.5	5.8	125.9	77.7	113.7	94.2	179.7	165.3	132.8	**227.7**
Assault	742.0	222.4	561.5	330.9	500.9	479.5	565.6	**702.3**	493.3	643.7
Asterix	8503.3	210.0	535.4	334.7	567.2	535.6	962.5	933.3	**1021.3**	883.7
Bank Heist	753.1	14.2	185.5	55.0	65.3	153.4	**345.4**	245.9	288.2	180.3
Battle Zone	37187.5	2360.0	8977.0	5139.4	8997.8	10563.6	14834.1	13260.0	13076.7	**16080.0**
Boxing	12.1	0.1	-0.3	1.6	0.9	6.6	35.7	**38.3**	14.3	26.4
Breakout	30.5	1.7	9.2	8.1	2.6	15.4	19.6	**20.6**	16.7	16.8
Chopper Cmd	7387.8	811.0	925.9	813.3	783.5	792.4	**946.3**	922.4	878.7	910.7
Crazy Climber	35829.4	10780.5	34508.6	14999.3	9154.4	21991.6	**36700.5**	23176.7	28235.7	24633.3
Demon Attack	1971.0	152.1	627.6	681.6	646.5	**1142.4**	517.6	1131.7	310.5	854.6
Freeway	29.6	0.0	20.9	11.5	28.3	17.8	19.3	16.1	**30.9**	30.2
Frostbite	4334.7	65.2	871.0	224.9	1226.5	508.1	1170.7	1984.7	994.3	**2381.1**
Gopher	2412.5	257.6	467.0	539.4	400.9	618.0	660.6	684.3	650.9	**822.3**
Hero	30826.4	1027.0	6226.0	5956.5	4987.7	3722.6	5858.6	**8597.5**	4661.2	7919.3
Jamesbond	302.8	29.0	275.7	88.0	331.0	251.8	366.5	394.7	270.0	**423.2**
Kangaroo	3035.0	52.0	581.7	348.5	740.2	974.5	3617.4	2384.7	5036.0	**8516.0**
Krull	2665.5	1598.0	3256.9	3655.9	3049.2	**4131.4**	3681.6	3880.7	3571.3	3923.1
Kung Fu Master	22736.3	258.5	6580.1	6659.6	8155.6	7154.5	**14783.2**	14259.0	10517.3	10652.0
Ms Pacman	6951.6	307.3	1187.4	908.0	1064.0	1002.9	1318.4	1335.4	1320.9	**1481.3**
Pong	14.6	-20.7	-9.7	-2.5	-18.5	-14.3	-5.4	-3.0	-3.1	**4.9**
Private Eye	69571.3	24.9	72.8	59.6	81.9	24.8	86.0	93.9	93.3	**100.0**
Qbert	13455.0	163.9	1773.5	552.5	727.0	934.2	866.3	**3620.1**	553.8	3410.4
Road Runner	7845.0	11.5	11843.4	2606.4	5006.1	8724.7	12213.1	**13429.4**	12337.0	12049.7
Seaquest	42054.7	68.4	304.6	272.9	315.2	310.5	558.1	532.9	471.9	**628.3**
Up N Down	11693.2	533.4	3075.0	2331.7	2646.4	3619.1	**10859.2**	10225.2	4112.8	6675.7
Interquartile Mean	1.000	0.000	0.183	0.117	0.113	0.224	0.337	0.374	0.292	**0.432**
Optimality Gap	0.000	1.000	0.698	0.819	0.768	0.692	0.577	0.558	0.614	**0.522**

Experiment

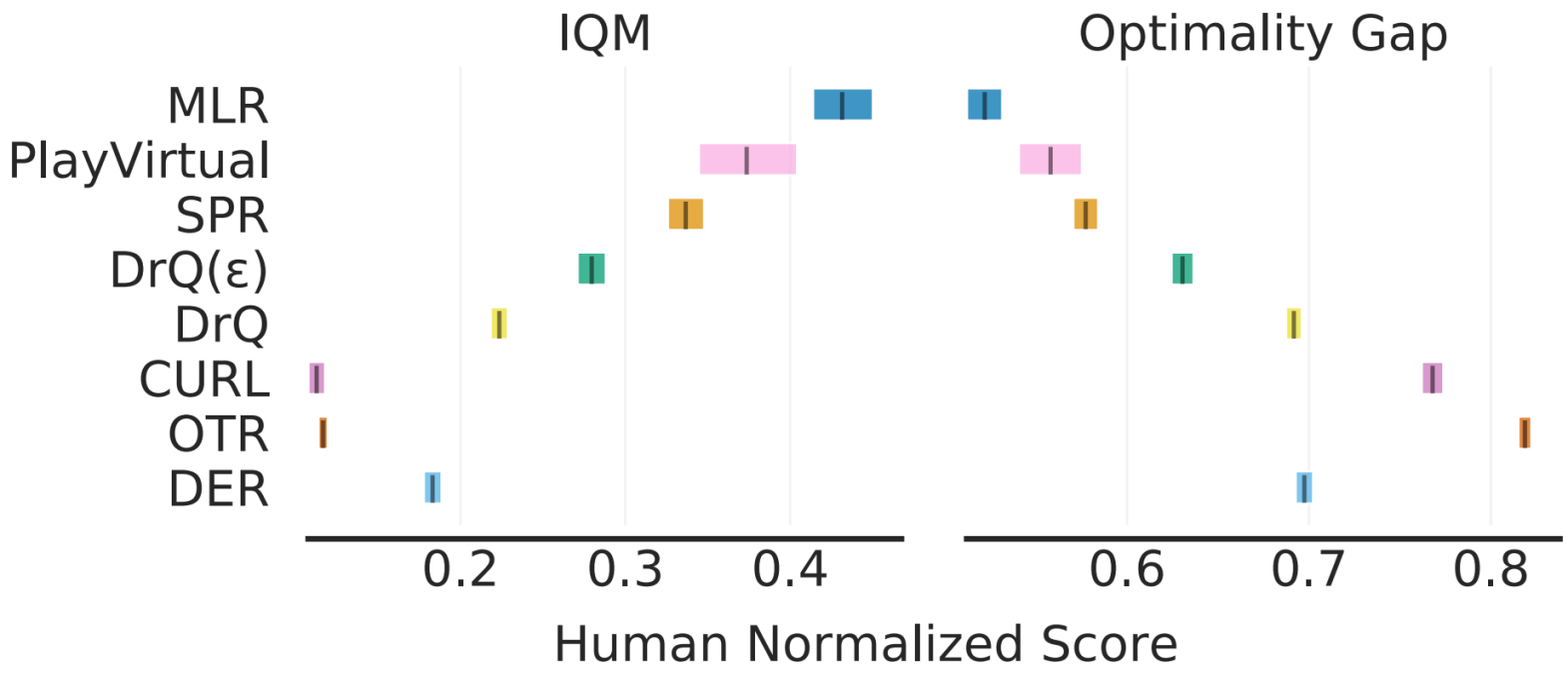


Figure 1: Comparison on the Atari-100k benchmark.

Experiment

Table 2: Comparison results (mean ± std) on the DMControl-100k and DMControl-500k benchmarks. Our method augments Baseline with the proposed MLR objective.

100k Step Scores	**PlaNet**	**Dreamer**	**SAC+AE**	**SLAC**	**CURL**	**DrQ**	**PlayVirtual**	**Baseline**	**MLR**
Finger, spin	136 ± 216	341 ± 70	740 ± 64	693 ± 141	767 ± 56	901 ± 104	**915 ± 49**	853 ± 112	907 ± 58
Cartpole, swingup	297 ± 39	326 ± 27	311 ± 11	-	582 ± 146	759 ± 92	**816 ± 36**	784 ± 63	806 ± 48
Reacher, easy	20 ± 50	314 ± 155	274 ± 14	-	538 ± 233	601 ± 213	785 ± 142	593 ± 118	**866 ± 103**
Cheetah, run	138 ± 88	235 ± 137	267 ± 24	319 ± 56	299 ± 48	344 ± 67	474 ± 50	399 ± 80	**482 ± 38**
Walker, walk	224 ± 48	277 ± 12	394 ± 22	361 ± 73	403 ± 24	612 ± 164	460 ± 173	424 ± 281	**643 ± 114**
Ball in cup, catch	0 ± 0	246 ± 174	391 ± 82	512 ± 110	769 ± 43	913 ± 53	926 ± 31	648 ± 287	**933 ± 16**
Mean	135.8	289.8	396.2	471.3	559.7	688.3	729.3	616.8	**772.8**
Median	137.0	295.5	351.0	436.5	560.0	685.5	800.5	620.5	**836.0**
500k Step Scores									
Finger, spin	561 ± 284	796 ± 183	884 ± 128	673 ± 92	926 ± 45	938 ± 103	963 ± 40	944 ± 97	**973 ± 31**
Cartpole, swingup	475 ± 71	762 ± 27	735 ± 63	-	841 ± 45	868 ± 10	865 ± 11	871 ± 4	**872 ± 5**
Reacher, easy	210 ± 390	793 ± 164	627 ± 58	-	929 ± 44	942 ± 71	942 ± 66	943 ± 52	**957 ± 41**
Cheetah, run	305 ± 131	570 ± 253	550 ± 34	640 ± 19	518 ± 28	660 ± 96	**719 ± 51**	602 ± 67	674 ± 37
Walker, walk	351 ± 58	897 ± 49	847 ± 48	842 ± 51	902 ± 43	921 ± 4575	928 ± 30	818 ± 263	**939 ± 10**
Ball in cup, catch	460 ± 380	879 ± 87	794 ± 58	852 ± 71	959 ± 27	963 ± 9	**967 ± 5**	960 ± 10	964 ± 14
Mean	393.7	782.8	739.5	751.8	845.8	882.0	**897.3**	856.3	896.5
Median	405.5	794.5	764.5	757.5	914.0	929.5	935.0	907.0	**948.0**

Experiment

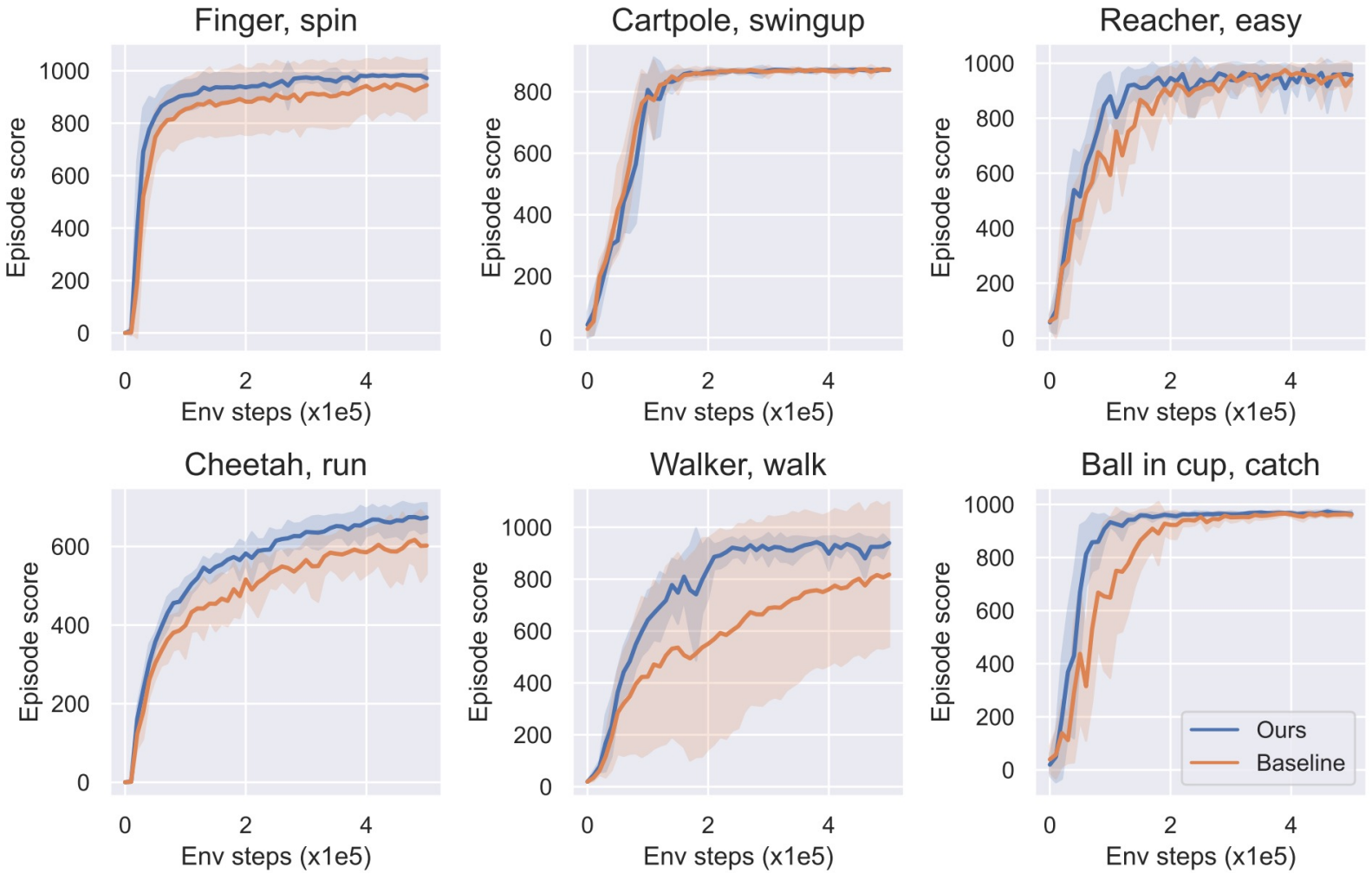


Figure 2: Comparison results on the DMControl benchmarks.

Thanks for Watching!

arXiv

Code