Molinaroli College of Engineering and Computing
UNIVERSITY OF SOUTH CAROLINA

Stable Planning through Aligned Representations in Model-Based Reinforcement Learning

Misagh Soltani   Forest Agostinelli

Department of Computer Science and Engineering, University of South Carolina

msoltani@email.sc.edu   foresta@cse.sc.edu

Code on GitHub

Download on the App Store

The Problem

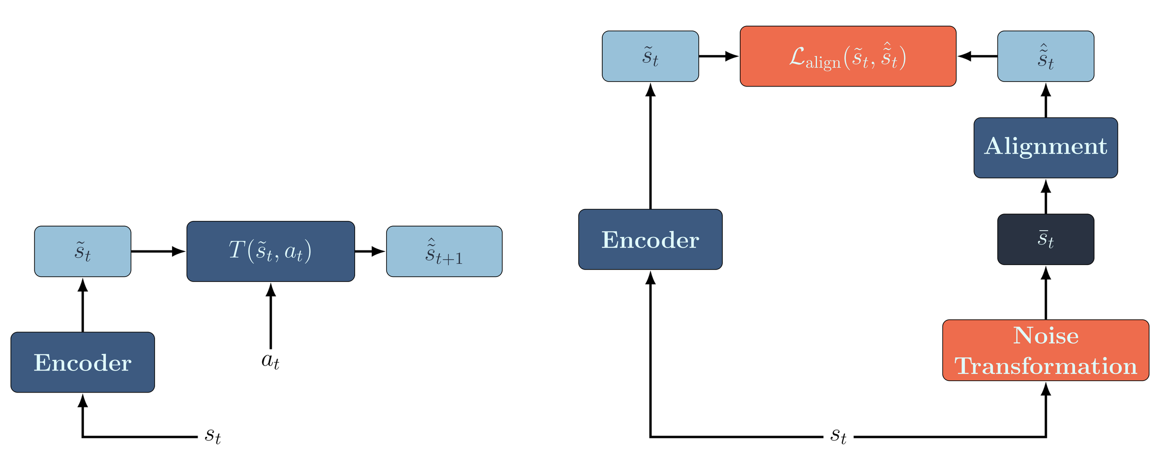
- Visual noise changes the latent state even when the underlying task is unchanged.
- Heuristic search needs **exact state re-identification** for duplicate detection and goal testing.
- New noise can make a goal image unreachable.

Why Not Train the World Model on Noisy Observations?

- Different noise produces different latent states, so exact state re-identification fails.
- New noise requires re-training the world model and heuristic function.
- Training on noisy observations captures noise-specific transitions and appearance variation.

SPAR

1. Train a world model with a **discrete latent space** on clean states.
2. Train an **alignment model** that maps noisy states to the same latent representation as the clean states.
3. **Round** the output of the alignment model so that it exactly matches the clean discrete latent state, then plan.
4. New noise requires **no retraining of the world model or heuristic function** and **no additional data collection**: adapt only the alignment model.



1 Discrete world model, trained on clean states

2 Alignment model, trained on noisy states

Neural Network | Predefined Process | Discrete Latent State | Noisy State

Discrete World Model

- The encoder E maps a state s to a discrete latent state $\bar{s} = E(s)$ through a logistic activation that is rounded to $\{0,1\}$, with a straight-through estimator enabling gradient-based training.
- The transition model T maps latent states and actions to next latent states.
- A reconstruction loss and a transition loss are combined,

$$\mathcal{L}(\theta) = (1-\omega)\,\mathcal{L}_{\text{rec}}(\theta) + \omega\,\mathcal{L}_{\text{m}}(\theta),$$

where ω starts small and is increased until both terms are weighted equally.
- Rounding the predicted latent state removes any error below 0.5, so the model does not degrade across thousands of timesteps.

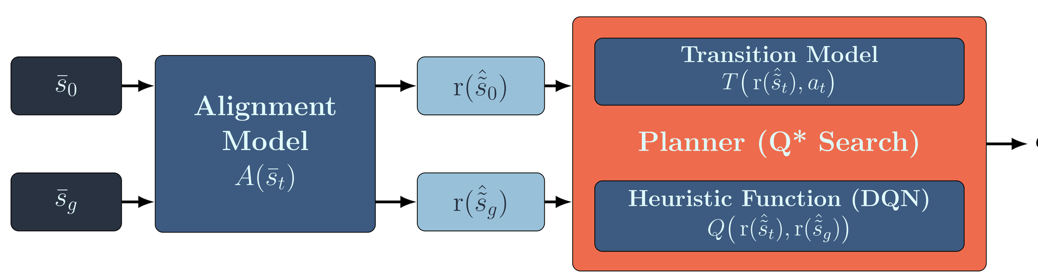
Alignment Model

- Noise is added to a clean state through a stochastic function $\mathcal{N}$, giving $\tilde{s}_t = \mathcal{N}(s_t)$, and the clean state is encoded to $\bar{s}_t = E(s_t)$.
- The alignment model A is trained with supervised learning to predict the clean discrete latent state from the noisy state,

$$\mathcal{L}_{\text{align}}(\phi) = \frac{1}{N}\sum_{i=1}^{N}\left\|A(\tilde{s}_t^{(i)}) - \bar{s}_t^{(i)}\right\|_2^2.$$

- At inference the outputs of A are rounded element-wise to $\{0,1\}$. If the error of every bit is below 0.5, the recovered latent state matches the true one exactly.
- The cost-to-go must be invariant to the noise (e.g., for the Rubik's Cube the noise may change the background color but not the color of the stickers).

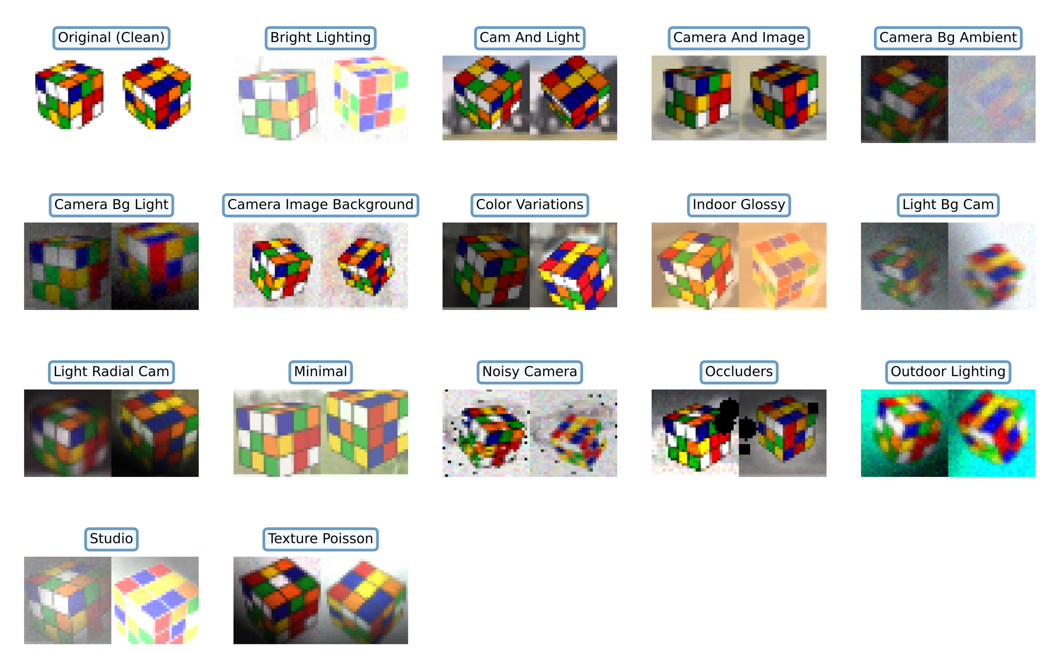
Finding Paths Given Noisy States

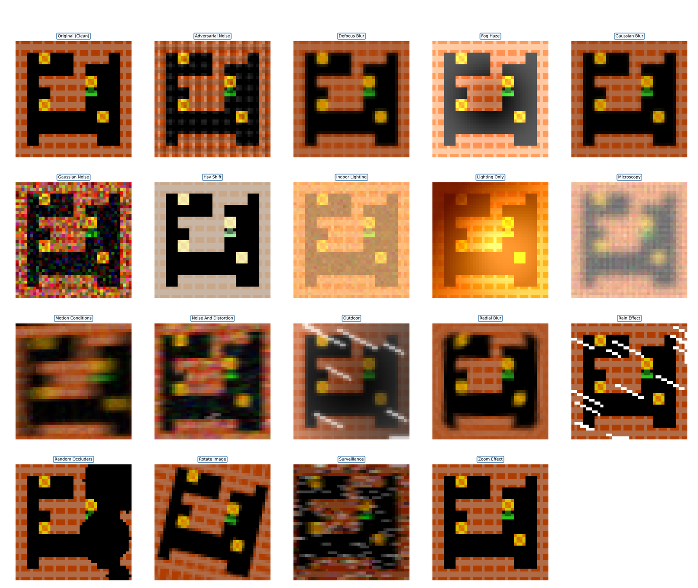


Noisy start and goal states are mapped to discrete latent states by the alignment model. Q* search then runs in latent space with the fixed world model and the learned DQN as the heuristic function, and the goal test compares all bits of the latent state.

Visual Transformations

The stochastic noise function is built from transformations to which the cost-to-go is invariant. Individual transformations are combined to form the **16** variants of the Rubik's Cube and the **18** variants of Sokoban.

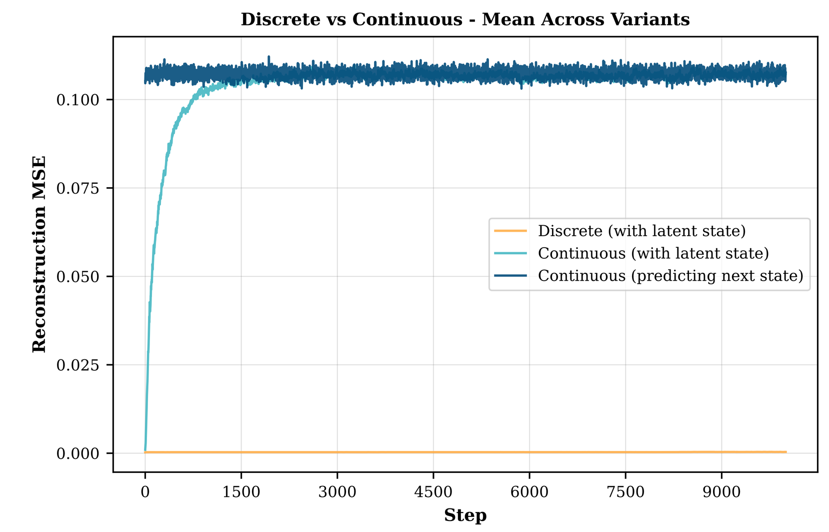




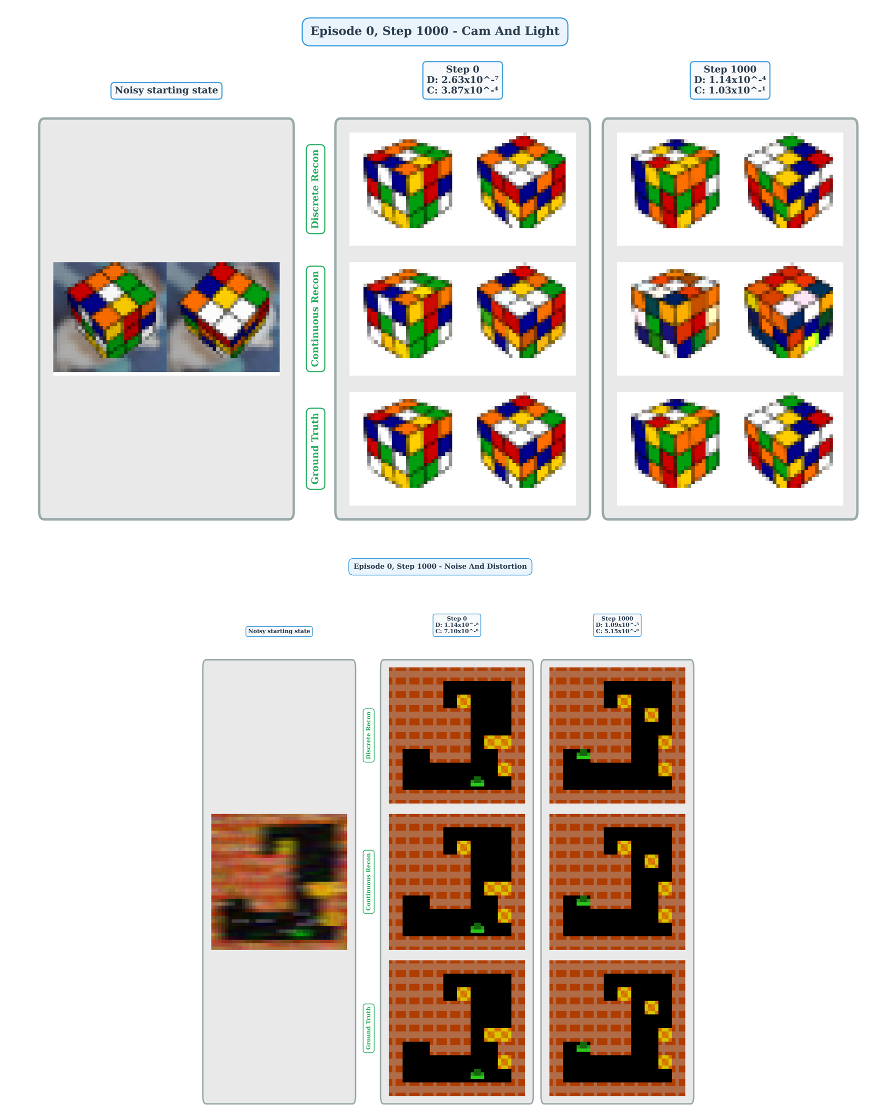
Examples of noisy alignment-model data from the Rubik's Cube and Sokoban domains.

Discrete Latents Prevent Model Degradation

Discrete latent states stay stable over long rollouts.



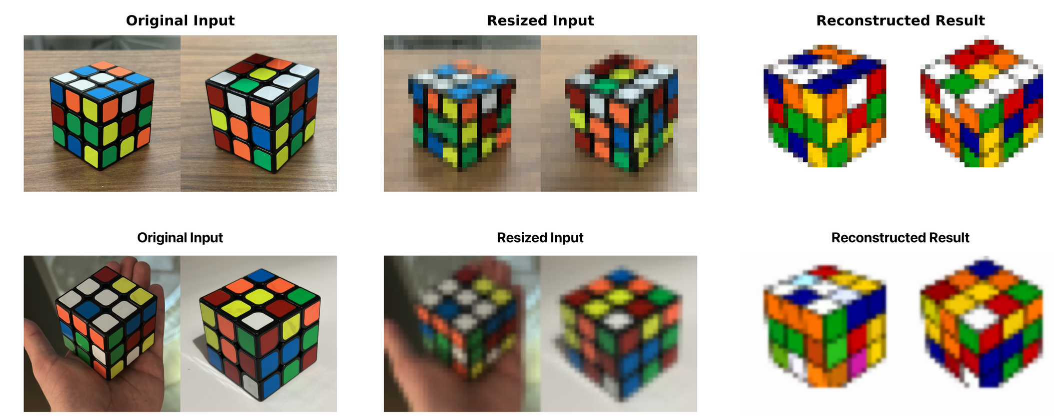
Reconstruction MSE averaged across the Rubik's Cube visual transformations.



Decoded output of the alignment model at the noisy start state and after 1,000 steps. The continuous model degrades, the discrete model does not.

Real-World Rubik's Cube

Examples of real-world photos taken under different environmental conditions (reconstruction on the output of the alignment model). **No real-world image was seen during training.** SPAR finds a solution path for instances that the alignment model maps to the same latent state as the clean start.

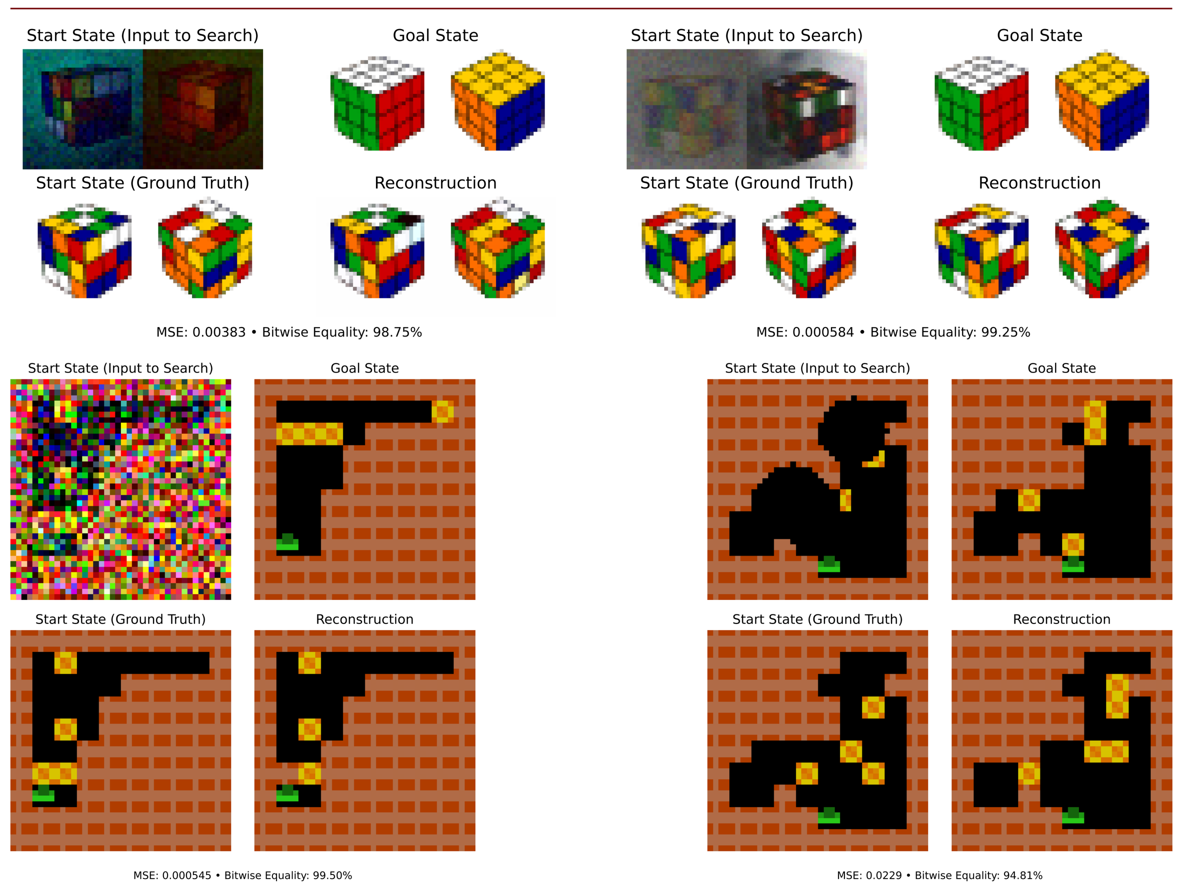


Planning Results

DOMAIN	OBSERVATION	SOLVER	LEN	NODES	SECS	SOLVED	95% CI
Rubik's Cube	Clean	SPAR	22.85	2.00E+05	6.21	100%	[99.6, 100.0]
		DeepCubeAI	22.85	2.00E+05	6.21	100%	[99.6, 100.0]
		Greedy Policy	–	–	–	0%	[0.0, 0.4]
	Noisy	SPAR	22.85	2.00E+05	6.21	**89.92%**	[89.4, 90.4]
		DeepCubeAI	–	–	–	0%	[0.00, 0.02]
		Greedy Policy	–	–	–	0%	[0.00, 0.02]
	Real-World	SPAR	23.61	1.90E+05	6.21	**50%**	[23.7, 76.3]
		DeepCubeAI	–	–	–	0%	[0.0, 27.8]
		Greedy Policy	–	–	–	0%	[0.0, 27.8]
Sokoban	Clean	SPAR	32.32	2.39E+04	20.58	100%	[99.6, 100.0]
		DeepCubeAI	32.32	2.39E+04	20.58	100%	[99.6, 100.0]
		Greedy Policy	–	–	–	41.9%	[38.9, 45.0]
	Noisy	SPAR	32.30	2.39E+04	20.55	**96.31%**	[96.0, 96.6]
		DeepCubeAI	–	–	–	0%	[0.00, 0.02]
		Greedy Policy	–	–	–	0%	[0.00, 0.02]

LEN is the length of the path found, NODES the number of nodes generated, and SECS the time to find a solution. Each visual variant is evaluated on 1,000 test instances, so the noisy rows aggregate 16,000 and 18,000 instances. The clean rows use 1,000 instances per domain and the real-world row uses 10 instances. DeepCubeAI was not designed for noisy observations and is included to show how a latent-space planner behaves without alignment.

Failure Cases



Failure cases show that noise can distort information needed to identify the state.

Take Aways & Future Work

- **Train once on clean states**: keep the world model and heuristic fixed, and adapt only the alignment model to new visual noise.
- **No real-world training data**: add noise to existing data. Rounding recovers exact discrete states when every bit error is below 0.5, preserving goal tests and long-rollout stability.
- **Results**: SPAR solves **89.92%** of noisy Rubik's Cube and **96.31%** of noisy Sokoban instances, and generalizes to phone photos without real-world training images.
- **Future: Robustness** Re-plan when expected observations diverge from actual ones, especially when noise occludes information needed to identify the state.
- **Future: Semantics and robotics** Use large vision models to link image features to latent bits, enabling formal logic goals and rapid perception recalibration under changing lighting and backgrounds.