A Momentum Transition Model for Complex Dynamic Environments in Reinforcement Learning

Ruihong Qiu, Lisen Mu, Zi Huang

School of ITEE, The University of Queensland, Australia

r.qiu@uqconnect.edu.au

THE UNIVERSITY OF QUEENSLAND AUSTRALIA

Problem

There are many advantages of model-based methods for reinforcement learning. However model-based methods still encounter limitations in many tasks where dynamics of environments are too complex to model. How to construct an accurate and efficient model for environments and use it for policy in RL remain challenging.

Motivation

- Rollout in image space is slow with redundant information while in state space, is efficient and compact.
- Separate approximations for state and action values are suitable for reinforcement learning.
- More efficient prediction of states can be integrated in an efficient way with algorithms.

Momentum Transition Model

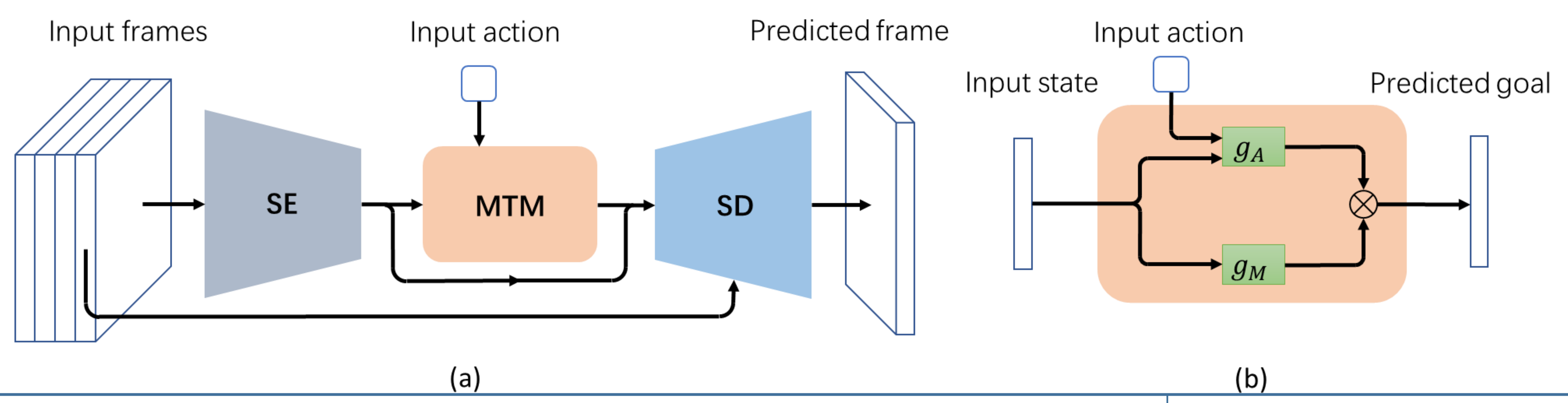

Figure 1.
(a) The pipeline of one-step prediction. A state encoder (SE) takes input observation. A Momentum Transition Model (MTM) predicts a current goal. A state decoder (SD) reconstructs the next frame.
(b) The separation inside MTM.

State Decoder

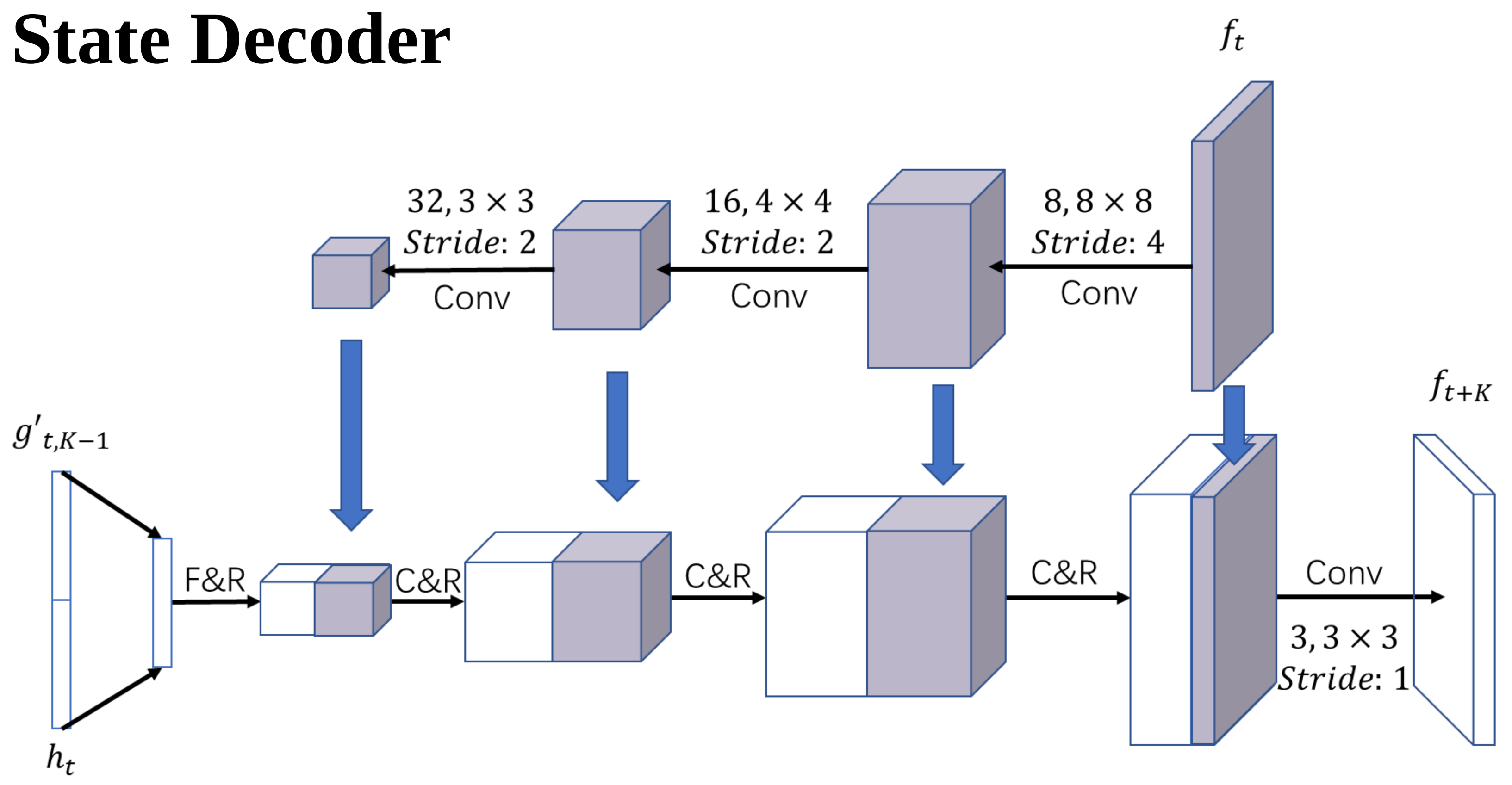

Figure 2. Structure of the state decoder (SD). The upper half is the encoding procedure of f_t. The input state h_t and the predicted goal $g'_{t,K-1}$ are concatenated together. $F\&R$ is an operation firstly applying fully-connected layer and then reshaping the result to a specific size while C&R applies a convolutional layer first.

Rollout

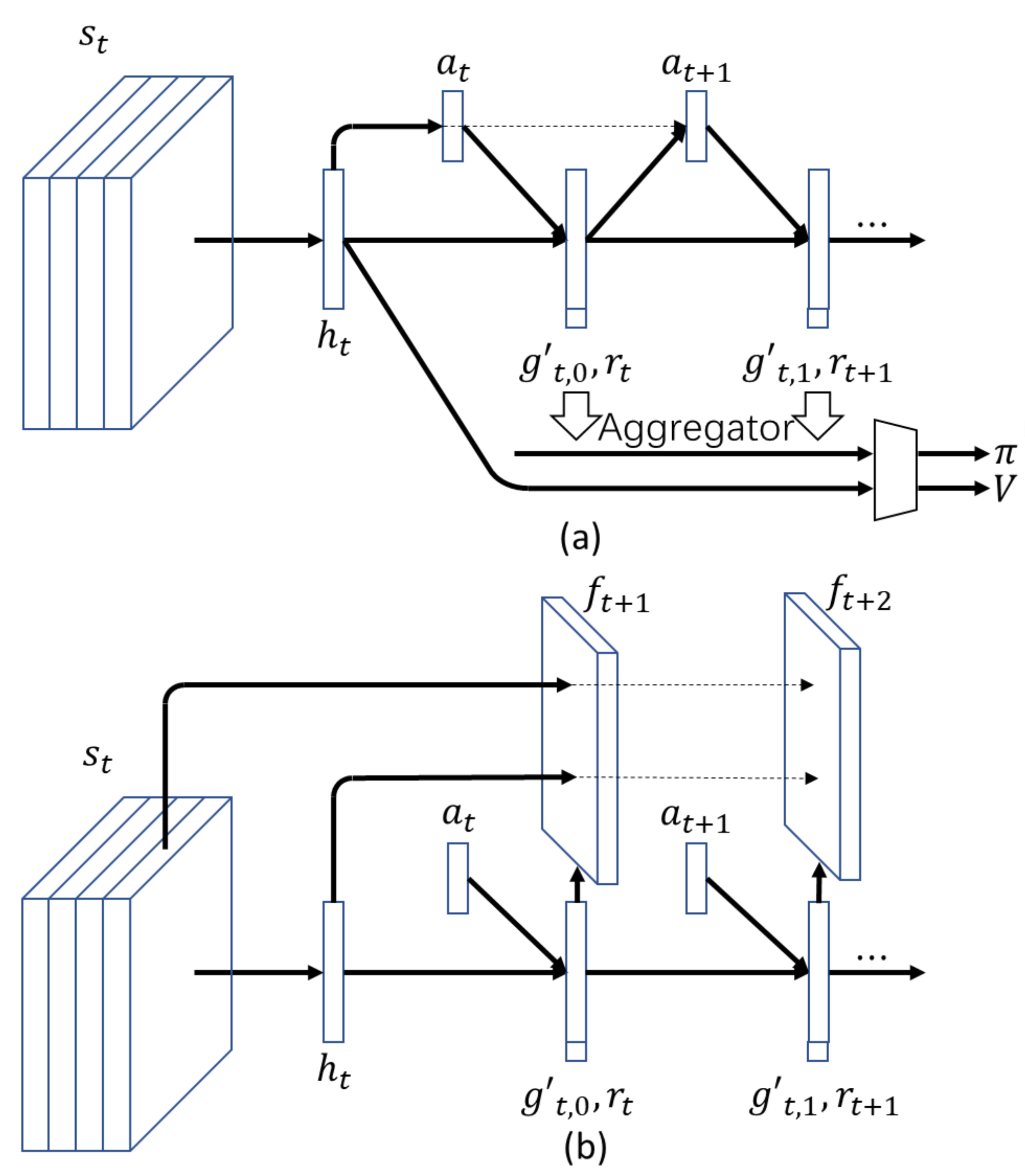

Figure 3.
(a) Rollout in policy in the form of I2A. The Aggregator encodes results of rollout and provides information for policy.
(b) Rollout in training with multi-step prediction. The only difference of the prediction at this different time step is the current goal. Goals give the hints for different dynamics of the environments and agents.

Experiments

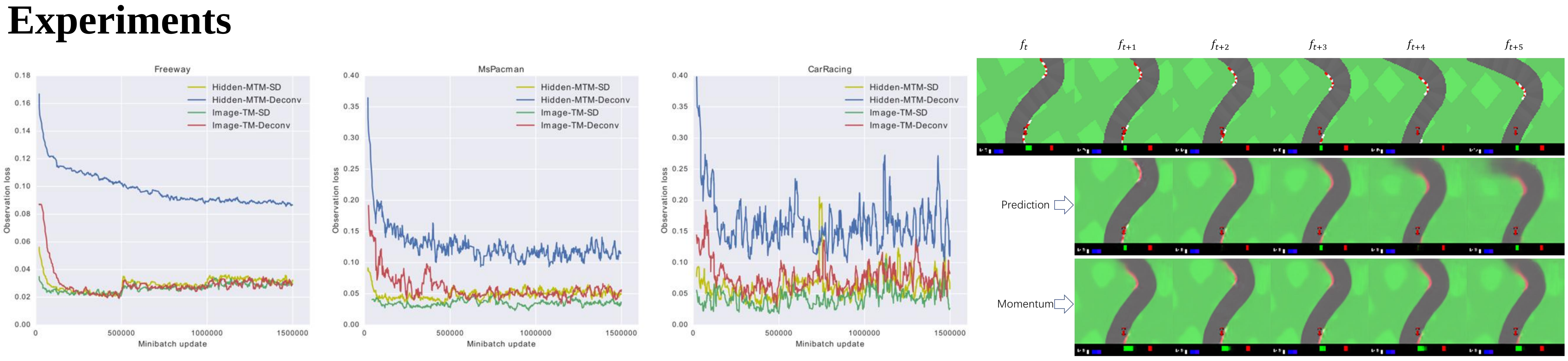

Figure 4. Visualization of how the momentum affects the prediction.

Conclusion

We present a novel method for building a model - MTM in deep RL. The MTM and SD architecture are effective respectively and when combined together. We build a model in a much lower dimension space but with similar representation ability in frames prediction. Our method is suitable for environments with system dynamics, which means that it is suitable for most video games. This approach can be further applied to control problems and real world environments. It has strong potential to be generalized to many other environments.