Information-Theoretic State Space Model for Multi-View Reinforcement Learning

HyeongJoo Hwang [1] **Seokin Seo** [1] **Youngsoo Jang** [2] **Sungyoon Kim** [1] **Geon-Hyeong Kim** [2]
Seunghoon Hong [3] **Kee-Eung Kim** [1 3]

Abstract

Multi-View Reinforcement Learning (MVRL) seeks to find an optimal control for an agent given multi-view observations from various sources. Despite recent advances in multi-view learning that aim to extract the latent representation from multi-view data, it is not straightforward to apply them to control tasks, especially when the observations are temporally dependent on one another. The problem can be even more challenging if the observations are intermittently missing for a subset of views. In this paper, we introduce Fuse2Control (F2C), an information-theoretic approach to capturing the underlying state space model from the sequences of multi-view observations. We conduct an extensive set of experiments in various control tasks showing that our method is highly effective in aggregating task-relevant information across many views, that scales linearly with the number of views while retaining robustness to arbitrary missing view scenarios.

1. Introduction

In real-world decision-making problems, the observation from the environment is often complex and high-dimensional. This is because the agent is usually equipped with multiple sensors to obtain a better sense of what's going on in the environment. For example, it is a standard practice to employ an array of sensors (e.g. lidars, cameras, sonars, etc.) in an autonomous vehicle. Treating each of these sensors as a view, Multi-View Reinforcement Learning (Li et al., 2019) (MVRL) aims to learn an optimal control policy in a complex control task based on multi-view observations. Although one could naively train an RL agent directly taking the stack of entire observations, it is unclear to handle incomplete observations from views that are intermittently missing, e.g. sensors operating at different frequencies or being occluded. Furthermore, it would be very sample-inefficient especially when observations are of high dimensions (Lake et al., 2017; Tassa et al., 2018; Kaiser et al., 2019). Consequently, it is necessary to adopt representation learning from multiple observations, i.e. Multi-View Learning (MVL), so that the agent learns to act based on a low-dimensional latent representation of the state, which retains robustness to missing views and provides sufficient information about the true state relevant to decision making.

Leveraging the power of deep generative models, MVL has made remarkable progress in recent years. Notably, multi-view generative models (Wu & Goodman, 2018; Shi et al., 2019; Sutter et al., 2020; Shi et al., 2019; Hwang et al., 2021) jointly train per-view Variational Autoencoders (VAEs) and obtain the latent representation by combining each representation from all views via weighted averaging strategies, such as Product of Experts (Wu & Goodman, 2018; Hwang et al., 2021), Mixture of Experts (Shi et al., 2019), or some more complex strategies (Sutter et al., 2020; 2021). These methods naturally extend to the partial-view scenario where an arbitrary subset of views could be missing per observation instance, while some of them are not computationally scalable for a large number of views. Recently, Hwang et al. (2021) showed that these approaches can be interpreted as optimizing the Total Correlation (TC) (Watanabe, 1960).

However, the direct application of these multi-view generative models to control tasks is not straightforward since we need to consider the sequential nature of the problem that arises from the dynamics of the environment. For example, it would be desirable to learn the latent state representation that exhibits Markov property for the sake of simplicity in the RL training loop. Furthermore, when some of the views are missing at a particular timestep, we should still be able to harness the available observation from previous time steps to infer the latent state.

In this paper, we propose Fuse2Control (F2C), a principled MVRL framework that discovers the underlying dynamics

[1]Kim Jaechul Graduate School of AI, KAIST, Daejeon, South Korea [2]LG AI Research, Seoul, South Korea [3]School of Computing, KAIST, Daejeon, South Korea. Correspondence to: HyeongJoo Hwang <hjhwang@ai.kaist.ac.kr>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

from multi-view observations and actions. Inspired by the recent MVL approach in (Hwang et al., 2021), we formulate the task of learning the latent state representation across multiple views as maximizing the informativeness measured in terms of TC, which reflects the multi-view aspect as well as the sequential nature of the problem. We show that a number of deep learning approaches for State Space Models (SSMs) are covered as special cases of our formulation, which facilitates understanding the problems associated with their direct application to MVRL. To this end, we derive an alternative lower bound that extends single-view (i.e. flat) SSMs to multi-view SSMs (MV-SSMs) by introducing Conditional Variational Information Bottlenecks (CVIBs) into our objective. Combined with the precision weighted averaging (Cochran & Carroll, 1953; Cochran, 1954), our method is shown to be effective in handling missing views, using computation that linearly scales with the number of views. Through experiments on various multi-view manipulation and locomotion tasks, we show that our method is not only sample-efficient in policy optimization, but also robust under various missing view scenarios.

2. Related Work

Data augmentation for learning representations in RL RAD (Laskin et al., 2020a) showed that applying a rich set of data augmentation techniques greatly accelerates learning an optimal policy. Following this work, there have been a number of self-supervised learning methods. For example, CURL (Laskin et al., 2020b), DRIBO (Fan & Li, 2022), and S2R (Yang et al., 2022) generate 2-view images by applying data augmentation techniques to the original image to isolate task-relevant information in the representation. While CURL minimizes contrastive loss between two views, DRIBO and S2R optimize Multi-view Information Bottleneck (Federici et al., 2020) and Conditional Entropy Bottleneck (Fischer, 2020) respectively. However, it is important to note that all these methods mainly aim to increase sample efficiency in one-view visual RL problems; all the experiment domains in CURL, DRIBO, and S2R are limited to two views where the second view is a result of image augmentation, a simple yet noisy transformation of the first view. Instead of generating redundant views, MvDAN (Hu et al., 2020) learns multiple policies or value functions that commonly use one-view observation and aggregates them with an attention module to solve single-view RL problems.

Learning dynamics model in RL By maximizing the log-likelihood of the trajectory data, state space models (SSMs) (Krishnan et al., 2015; Lee et al., 2020a) or its recurrent variants (Hafner et al., 2019; 2020; 2021) explicitly learn latent dynamics of the environment. These are closely related to ours, which we will discuss in Section 4.2.

Multi-view learning Multi-view generative models (Wu & Goodman, 2018; Shi et al., 2019; Sutter et al., 2020; 2021; Hwang et al., 2021) are one of the main approaches to MVL, jointly training multiple single-view VAEs. Based on their structural assumptions of the joint representation to be a weighted average of per-view representations, these methods can be trained even with an incomplete dataset where views are arbitrarily missing. To encourage alignment of per-view representations, MVTCAE (Hwang et al., 2021) regularizes the joint representation to be evenly dependent on all views while calibrating per-view representation encoders using CVIBs, which also plays a key role in our MV-SSM. As another dominant approach, CMC (Tian et al., 2020) and GMC (Poklukar et al., 2022) optimize the contrastive loss. Since CMC learns per-view representations without joint representation, it can be trained with missing-view data by minimizing the contrastive loss between every pair of available views, although this makes the optimization scale in quadratic with the number of views. In contrast, GMC learns the joint representation by minimizing the loss between joint representation and each of the per-view representations. However, it does not support a straightforward extension to missing-view scenarios since the joint representation requires all views to be available during training.

Multi-View RL Based on VAE architectures, Li et al. (2019) proposed a model to address MVRL. Without the notion of a joint state, the model minimizes the Euclidean distance between the encoded state from the first view and those from the rest. As such, it treats the first view as a primary view which is assumed to be always available. In contrast, Keypoint3D (Chen et al., 2021) learns 3d visual keypoints from multiple third-person view cameras, regularizing those keypoints to satisfy geometric constraints imposed by the configuration of cameras. While it addresses MVRL, it can be only applied to third-person camera views whose calibration parameters are required to be known a-priori. On the other hand, LookCloser (Jangir et al., 2022) addresses MVRL using 2 images, one from an egocentric and the other from the third-person-view camera, without calibration parameters. LookCloser adopts cross-view attention for aggregating per-view representations. Although one can extend it to more than 2 views by applying cross-view attention to all pairs of views, its computation would scale quadratically with the number of views.

3. Background

Total Correlation in Multi-View Learning The Total Correlation (TC) (Watanabe, 1960) is one of the representative measures of dependency among a set of random variables (RVs). It is defined as the Kullback-Leibler divergence between the joint distribution of the RVs and the product of their marginal distributions. In the context of MVL with V

observations $\vec{o} = \{o^v\}_{v=1}^V$ obtained from an unknown joint distribution $p_D(\vec{o})$, the TC of observations is defined as

$$TC(\vec{O}) \triangleq D_{KL}\Big[p_D(\vec{o}) \| \prod_{v=1}^V p_D\left(o^v\right)\Big].$$

For a latent representation Z encoded by a stochastic encoder $p_\theta\left(z|\vec{o}\right)$, we measure the informativeness of Z by quantifying the reduction of the dependency among o^v's by conditioning on the latent representation Z. Formally, it is measured by the difference between TC and conditional TC,

$$TC_\theta(\vec{O}; Z) \triangleq TC(\vec{O}) - TC_\theta(\vec{O} \mid Z), \tag{1}$$

where the $TC_\theta(\vec{O} \mid Z)$ is the expected Kullback-Leibler divergence of the joint conditional from the factored conditionals (Hwang et al., 2021), defined as

$$TC_\theta(\vec{O}|Z) \triangleq \mathbb{E}_{p_\theta(z)}\Big[D_{KL}\Big[p_\theta\left(\vec{o}|z\right) \| \prod_{v=1}^V p_\theta\left(o^v|z\right)\Big]\Big].$$

Note that the above formula involves encoder $p_\theta\left(z|\vec{o}\right)$ such that $p_\theta\left(z\right) = \int p_\theta\left(z|\vec{o}\right) p_D\left(\vec{o}\right) d\vec{o}$, $p_\theta\left(\vec{o}|z\right) = p_\theta\left(z|\vec{o}\right) p_D\left(\vec{o}\right)/p_\theta\left(z\right)$, and $p_\theta\left(o^v|z\right) = \int p_\theta\left(\vec{o}|z\right) d\vec{o}^{\setminus v}$.

If $p_\theta\left(z|\vec{o}\right)$ completely captures all the factors of variation in $\vec{o}$ so that it encodes *complete representation* Z, $TC_\theta(\vec{O}; Z)$ would be maximized since any complete representation factorizes $p_\theta(\vec{o}|z)$ and thus makes the second term in Eq. (1) vanish (Ver Steeg & Galstyan, 2015; Gao et al., 2019; Ver Steeg & Galstyan, 2014; Hwang et al., 2021).

Ver Steeg & Galstyan (2014) observed that $TC_\theta(\vec{O}; Z)$ can be alternatively expressed as a decomposition into multiple Mutual Information (MI) terms,

$$TC_\theta(\vec{O}; Z) = \textstyle\sum_{v=1}^V I_\theta\left(O^v; Z\right) - I_\theta(\vec{O}; Z), \tag{2}$$

where $I_\theta(\vec{O}; Z)$ is known as Information Bottleneck (IB) (Tishby et al., 2000). To learn a balanced representation that is evenly dependent on views, Hwang et al. (2021) derived a variational lower bound introducing Conditional Variational IBs (CVIBs).

$$\begin{aligned} TC_\theta(\vec{O}; Z) &= \tfrac{1}{V} \sum_{v=1}^V \Big[(V \text{ - } 1)\, I_\theta\left(O^v; Z\right) \text{ - } I_\theta(\vec{O}^{\setminus v}; Z|O^v)\Big] \\ &\geq \tfrac{V-1}{V} \sum_{v=1}^V \left[H\left(O^v\right) + \mathbb{E}_{p_\theta(z,\vec{o})}\left[\ln q_\phi^v\left(o^v|z\right)\right]\right] \\ &\quad - \tfrac{1}{V} \sum_{v=1}^V \underbrace{\mathbb{E}_{p_D(\vec{o})}\left[D_{KL}\left[p_\theta(z|\vec{o}) \| r_\psi^v(z|o^v)\right]\right]}_{\text{CVIB}}. \end{aligned} \tag{3}$$

MV-POMDP We model the sequential decision-making problem under multi-view observations as a Multi-View Partially Observable Markov Decision Process (MV-POMDP) (Kaelbling et al., 1998), defined by tuple $\left\langle \mathcal{S}, \mathcal{A}, \vec{\mathcal{O}}, T, \Omega, R, \gamma, p_0 \right\rangle$, where $\mathcal{S}$ is the set of unknown ground-truth states s, $\mathcal{A}$ is the set of actions a, $\vec{\mathcal{O}} = \{\mathcal{O}^v\}_{v=1}^V$ is the set of V observations $\vec{o} = \{o^v\}_{v=1}^V$, $T(s'|s,a) = \Pr(s_{t+1} = s'|s_t = s, a_t = a)$ is the transition dynamics distribution, $\Omega(\vec{o}|s) = \prod_{v=1}^V \Pr(o_t^v = o^v|s_t = s)$ is the joint observation probability distribution, $R(s,a) \in \mathbb{R}$ is the immediate reward function for taking action a at state s, $\gamma \in [0,1)$ is the discount factor, and $p_0(s) = \Pr(s_0 = s)$ is the starting state distribution at timestep 0. Figure 1 describes the graphical model that represents the MV-POMDP. Given an unknown data-collecting policy $\pi_D(a_t|\vec{o}_{\leq t})$ that is dependent on the history of observations, complete-view trajectories can be collected from the following distribution.

$$\begin{aligned} &p_D(a_{\leq T}, \vec{o}_{\leq T}) \\ &= \int p_0(s_0) \prod_{t=0}^T \Omega(\vec{o}_t|s_t) \pi_D(a_t|\vec{o}_{\leq t}) T(s_{t+1}|s_t, a_t) ds_{\leq T}. \end{aligned}$$

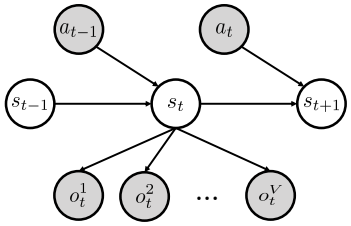


Figure 1. Graphical model of MV-POMDP.

4. Method

Multi-View Reinforcement Learning (MVRL) is the problem of learning an optimal policy in an environment modeled as an MV-POMDP. Since the ground-truth states are not available, the agent has to infer the state from the history of observations and actions, known as the *belief state*. Ideally, the belief state should summarize all the information up to the present about the ground-truth state. This Markov property is essential for many off-the-shelf RL methods to work properly. The main challenge here is to accurately infer the belief state while observations from a subset of views could be arbitrarily missing in each timestep. This is compounded by the fact that some observations from views may be of very high dimensions, such as raw images from cameras. At the same time, we want the belief state to be of low dimension to tame the complexity inside the RL loop.

Thus, we aim to learn the belief state encoding that is not only minimal yet sufficiently informative about the ground-truth state but also robust to missing views. For this, we first derive an objective function based on TC that carefully relates states, actions, successor states, and multi-view observations for each timestep (Section 4.1). We then show a close connection between our formulation and well-known (single-view) RL methods that learn state space models (SSMs), which reveals problems associated when

naively used for incomplete (missing-view) observations (Section 4.2). We derive an alternative objective function that extends SSMs to Multi-View SSMs (MV-SSMs), which admits optimization that scales linearly with the number of views and handles missing views effectively (Section 4.3). Lastly, we finalize our framework by describing the overall procedure for training MV-SSM and the RL policy (Section 4.4).

4.1. Information Theoretic Approach to MVRL

To learn a stochastic encoder p_θ for the latent state $\hat{s}$, it is essential to investigate the relationship among the state, the action, the successor state, and the multi-view observations in two consecutive timesteps. As shown in Fig 1, the state $s_{t\text{-}1}$ and the action $a_{t\text{-}1}$ in the previous timestep $(t\text{-}1)$ jointly determine the successor state s_t, which yields new observations $\vec{o}_t = \{o_t^v\}_{v=1}^V$. From this generative flow behind the graphical model, we remark two important properties involving the ground state s_t:

1. Treating $\langle S_{t\text{-}1}, A_{t\text{-}1}\rangle$ as one joint random variable, $\langle s_{t\text{-}1}, a_{t\text{-}1}\rangle$ affects the generation of $\{o_t^v\}_{v=1}^V$ and thus $\langle s_{t\text{-}1}, a_{t\text{-}1}\rangle, o_t^1, ..., o_t^V$ are dependent.
2. Given s_t, however, $\vec{o}_t = \{o_t^v\}_{v=1}^V$ are solely dependent on s_t. Consequently, $\langle s_{t\text{-}1}, a_{t\text{-}1}\rangle, o_t^1, ..., o_t^V$ are conditionally independent on one another.

Given a stochastic belief state encoder $p_\theta\left(\hat{s}_t|\vec{o}_t, \hat{s}_{t\text{-}1}, a_{t\text{-}1}\right)$ and samples from the fixed and unknown data distribution $o_{\leq t}, a_{<t} \sim p_D(\cdot)$, these two properties are naturally satisfied for the latent state $\hat{s}_t$ replacing the ground state s_t by maximizing $TC_\theta(\langle \hat{S}_{t\text{-}1}, A_{t\text{-}1}\rangle, \vec{O}_t; \hat{S}_t)$, given by

$$\begin{aligned} &TC_\theta(\langle \hat{S}_{t\text{-}1}, A_{t\text{-}1}\rangle, \vec{O}_t; \hat{S}_t) \\ &= TC_\theta(\langle \hat{S}_{t\text{-}1}, A_{t\text{-}1}\rangle, \vec{O}_t) - TC_\theta(\langle \hat{S}_{t\text{-}1}, A_{t\text{-}1}\rangle, \vec{O}_t \mid \hat{S}_t). \end{aligned} \tag{4}$$

Intuitively, this is because the maximization of the first term, $TC_\theta(\langle \hat{S}_{t\text{-}1}, A_{t\text{-}1}\rangle, \vec{O}_t)$, maximizes the dependency among all the current observations and the previous state-action pair, i.e. we want the belief state encoder to convey as much information as possible from the previous state-action pair, while the minimization of the second term, $TC_\theta(\langle \hat{S}_{t\text{-}1}, A_{t\text{-}1}\rangle, \vec{O}_t \mid \hat{S}_t)$, makes them independent when $\hat{S}_t$ is observed, i.e. a Markovian transition.

Treating the state-action pair in the previous timestep $\langle \hat{S}_{t\text{-}1}, A_{t\text{-}1}\rangle$ as an augmented view, Eq. (4) is apparantly analogous to Eq. (1). However, unlike the first term in TC being constant in Eq. (1), we need to optimize $TC_\theta(\langle \hat{S}_{t\text{-}1}, A_{t\text{-}1}\rangle, \vec{O}_t)$ since it involves learning the belief state encoder in the previous timestep.

Under mild assumptions on the data-collecting policy π_D, we further observe that the optimal solution of our objective function yields the latent state that is sufficient for optimal control (Li et al., 2006; Rakelly et al., 2021). More formally, denoting the optimal policy $\pi^* = \arg\max_{\pi\in\Pi} \mathbb{E}_\pi[\sum_{i=1}^H \gamma^{i-1} r(s_i)]$ and optimal Q-function $Q^* = Q^{\pi^*}$, where Π is the set of stationary policies, we can show that the following theorem:

Theorem 1. Let tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, T, R, \gamma, p_0\rangle$ be the underlying MDP in MV-POMDP. Assuming that data-collecting policy π_D is *ergodic* and has full support on $\mathcal{A}$, if the belief state encoder $p_\theta\left(\hat{s}_t|\vec{o}_t, \hat{s}_{t\text{-}1}, a_{t\text{-}1}\right)$ maximizes $\forall t > 0, TC_\theta(\langle \hat{S}_{t\text{-}1}, A_{t\text{-}1}\rangle, \vec{O}_t; \hat{S}_t)$ (Eq. (4)), then the state representation $\hat{\mu}_{\pi_D}(\mathrm{s}) = \lim_{t\to\infty} p_\theta(\hat{s}_t|s_t = \mathrm{s})$ is Q^*-sufficient for control. In other words, if any two states $\mathrm{s}^{(1)}, \mathrm{s}^{(2)} \in \mathcal{S}$ are mapped into the same representation such that $\hat{\mu}_{\pi_D}(\mathrm{s}^{(1)}) = \hat{\mu}_{\pi_D}(\mathrm{s}^{(2)})$, their Q^*-values are identical across all actions.

Proof. See Section A in the supplementary material. □

In order to align the optimization of our objective function with our theoretical observation, we adopt the assumption made by Sermanet et al. (2018) that complete-view data is available during the (pre)training of the representation. However, we do not assume any pattern of the missing-view observation $\tilde{o}_t \subseteq \vec{o}_t$ provided in test time (e.g. training policy, inferring missing views). Before we present the technical details of the optimization, we make connections to some well-known model learning methods for RL.

4.2. Connection to State Space Models (SSMs)

First, we rewrite $TC_\theta(\langle \hat{S}_{t\text{-}1}, A_{t\text{-}1}\rangle, \vec{O}_t; \hat{S}_t)$ in terms of MI:

$$\begin{aligned} TC_\theta(\langle \hat{S}_{t\text{-}1}, A_{t\text{-}1}\rangle, \vec{O}_t; \hat{S}_t) &= I_\theta(\hat{S}_{t\text{-}1}, A_{t\text{-}1}; \hat{S}_t) \\ &\quad + \sum_{v=1}^V I_\theta(O_t^v; \hat{S}_t) - I_\theta(\hat{S}_{t\text{-}1}, A_{t\text{-}1}, \vec{O}_t; \hat{S}_t) \quad (5) \\ &= \sum_{v=1}^V I_\theta(O_t^v; \hat{S}_t) - I_\theta(\vec{O}_t; \hat{S}_t \mid \hat{S}_{t\text{-}1}, A_{t\text{-}1}), \quad (6) \end{aligned}$$

where the equality in Eq. (5) follows from Eq. (2) and the equality in Eq. (6) holds due to the chain rule for MI (see Section B.1 and Section B.2 in the supplementary material). Next, we introduce q_ϕ^v and r_ψ^0 to derive a variational lower bound on Eq. (6) since the MI terms involve intractable integrals (see Section B.3 for detailed derivation):

$$\begin{aligned} &TC_\theta(\langle \hat{S}_{t\text{-}1}, A_{t\text{-}1}\rangle, \vec{O}_t; \hat{S}_t) \\ &\geq \sum_{v=1}^V \left[H\left(O_t^v\right) + \mathbb{E}_{p_\theta(o_t^v, \hat{s}_t)}\left[\ln q_\phi^v\left(o_t^v|\hat{s}_t\right)\right]\right] \quad (7) \\ &\quad - \mathbb{E}\left[D_{KL}\left[p_\theta\left(\hat{s}_t|\vec{o}_t, \hat{s}_{t\text{-}1}, a_{t\text{-}1}\right) \| r_\psi^0\left(\hat{s}_t|\hat{s}_{t\text{-}1}, a_{t\text{-}1}\right)\right]\right], \end{aligned}$$

where $p_\theta(o_t^v, \hat{s}_t) = \int p_D\left(\vec{o}_{\leq t}, a_{<t}\right) p_\theta\left(\hat{s}_0 \mid \vec{o}_0\right) \prod_{t'=1}^t p_\theta\left(\hat{s}_{t'}|\vec{o}_{t'}, \hat{s}_{t'\text{-}1}, a_{t'\text{-}1}\right) d\vec{o}_t^{\setminus v} d\vec{o}_{<t} d\hat{s}_{<t} da_{<t}$ and

the expectation in the last term is with respect to $p_\theta(\vec{o}_t, \hat{s}_{t-1}, a_{t-1}) = \int p_D(\vec{o}_{\le t}, a_{<t})\, p_\theta(\hat{s}_0 \mid \vec{o}_0) \prod_{t'=1}^{t-1} p_\theta(\hat{s}_{t'} \mid \vec{o}_{t'}, \hat{s}_{t'-1}, a_{t'-1})\, d\vec{o}_{<t} d\hat{s}_{<t-1} da_{<t-1}$. Eq. (7) naturally introduces the observation model $q_\phi^v(o_t^v \mid \hat{s}_t)$ for v-th view and the state transition model $r_\psi^0(\hat{s}_t \mid \hat{s}_{t-1}, a_{t-1})$.

Interestingly, Eq. (7) shows that recent world models in (single-view) RL that learn State Space Models (SSMs) and Recurrent SSMs (RSSMs) can be interpreted as special cases of our TC objective:

1. Ignoring the constant entropy terms $H(O_t^v)$ and assuming single-view observation ($V = 1$), Eq. (7) exactly reduces to the ELBO objective for SSMs (Krishnan et al., 2015; Lee et al., 2020a).
2. In addition, casting the history of states and actions as the augmented view $\langle \hat{S}_{t-1}, A_{t-1} \rangle = (\hat{S}_{<t}, A_{<t})$, Eq. (7) reduces to the ELBO objective for the generative model analogous to RSSMs (Hafner et al., 2019; 2020; 2021), with a minor difference in the decoder (see Section C.1 in the supplementary material).

Yet, a direct application of single-view SSM (and RSSM) to the multi-view setting is non-trivial since (1) the state encoder would require *complete-view* observations from all sensors (views) $\vec{o}$ for training and inference, and (2) the state encoder may end up learning to *ignore* less informative views, which can be critical when some observations become missing during inference. We address these issues by deriving an alternative objective in the next section.

4.3. Multi-View State Space Model (MV-SSM)

Objective To be able to handle missing-view observations and evenly allocate dependency across views in our objective, we rewrite Eq. (5) by applying the chain rule for MI (see Section B.2 for details).

$$
\begin{aligned}
&TC_\theta(\langle \hat{S}_{t-1}, A_{t-1} \rangle, \vec{O}_t; \hat{S}_t) \qquad (8)\\
&= \tfrac{1}{V+1} \sum_{v=1}^{V} [V \cdot \underbrace{I_\theta(O_t^v; \hat{S}_t)}_{\star} - \underbrace{I_\theta(\hat{S}_{t-1}, A_{t-1}, \vec{O}_t^{\setminus v}; \hat{S}_t | O_t^v)}_{\star\star}]\\
&\quad + \tfrac{1}{V+1} \big(V \cdot \underbrace{I_\theta(\hat{S}_{t-1}, A_{t-1}; \hat{S}_t)}_{\star\star\star} - \underbrace{I_\theta(\vec{O}_t; \hat{S}_t | \hat{S}_{t-1}, A_{t-1})}_{\star\star\star\star}\big)
\end{aligned}
$$

Compared to Eq. (6), this formulation explicitly introduces V conditional MIs ($\star\star$) between the state and previous state-action pair augmented with V-1 other observations given each observation, while maintaining important terms such as MI ($\star$) between every observation and the latent state, and the conditional MI ($\star\star\star\star$) between the latent state and V observations given the previous state-action pair. The new conditional MI terms penalize encoding information not inferable from the given view, which are upper-bounded by CVIBs that effectively regularize the joint latent state to be evenly dependent on views (Hwang et al., 2021).

Eq. (8) also introduces MI ($\star\star\star$) between the previous state-action pair and the current latent state which is non-trivial to bound. This is because the variational formulation of $I_\theta(\hat{S}_{t-1}, A_{t-1}; \hat{S}_t)$ would involve reconstruction error of latent variables that cannot be measured sensibly (Section B.5 for detailed information). To this end, we adopt Noise Contrastive Estimation (NCE) (Oord et al., 2018) which is known to lower-bound the MI between two RVs with low variance (Poole et al., 2019). NCE loss is defined as

$$
\begin{aligned}
&I_\theta(\hat{S}_{t-1}, A_{t-1}; \hat{S}_t) \qquad (9)\\
&\ge \mathbb{E}\left[\frac{1}{K}\sum_{i=1}^{K} \log \frac{e^{g(\hat{s}_{t-1}^{(i)}, a_{t-1}^{(i)}, \hat{s}_t^{(i)})}}{\frac{1}{K}\sum_{j=1}^{K} e^{g(\hat{s}_{t-1}^{(j)}, a_{t-1}^{(j)}, \hat{s}_t^{(i)})}}\right] \triangleq I_{\text{NCE}}(t; \theta),
\end{aligned}
$$

where K is the batch size, the function g is a neural network jointly optimized with states, and the expectation is with respect to $\prod_{k=1}^{K} p_\theta(\hat{s}_{t-1}^{(k)}, a_{t-1}^{(k)}, \hat{s}_t^{(k)})$.

MV-SSM Using the I_{NCE} objective, we are now ready to derive a variational lower bound of Eq. (8) to generalize SSMs to multi-view observations, given as

$$
\begin{aligned}
&TC_\theta(\langle \hat{S}_{t-1}, A_{t-1} \rangle, \vec{O}_t; \hat{S}_t) \ge \tfrac{V}{V+1} I_{\text{NCE}}(t; \theta)\\
&\quad + \tfrac{V}{V+1} \sum_{v=1}^{V} \left[H(O_t^v) \mathbb{E}_{p_\theta(o_t^v, \hat{s}_t)} \left[\ln q_\phi^v(o_t^v | \hat{s}_t)\right]\right]\\
&\quad - \tfrac{1}{V+1} \mathbb{E}\left[D_{KL}\left[p_\theta(\hat{s}_t | \vec{o}_t, \hat{s}_{t-1}, a_{t-1}) \,\|\, r_\psi^0(\hat{s}_t | \hat{s}_{t-1}, a_{t-1})\right]\right]\\
&\quad - \tfrac{1}{V+1} \sum_{v=1}^{V} \mathbb{E}\left[D_{KL}\left[p_\theta(\hat{s}_t | \vec{o}_t, \hat{s}_{t-1}, a_{t-1}) \,\|\, r_\psi^v(\hat{s}_t | o_t^v)\right]\right]\\
&= TC_{\text{MV-SSM}}(t; \theta, \phi, \psi), \qquad (10)
\end{aligned}
$$

where the expectations in the last two terms are with respect to $p_\theta(\vec{o}_t, \hat{s}_{t-1}, a_{t-1})$. Each of V+1 CVIB terms (Hwang et al., 2021) in Eq. (10) introduces and calibrates a state encoder $r_\psi^v(\hat{s}_t \mid o_t^v)$ for each view (see Section B.4 for detailed derivation). Since the CVIB of each view is the forward KL divergence between the joint state encoder and the per-view state encoder, its optimization encourages the per-view encoder to cover all the supports of the joint encoder. Hwang et al. (2021) demonstrated that this enables each per-view encoder to infer the joint latent state by capturing the relevant factors of variation observed from the given view, while maintaining uncertainty about the unobservable factors. In addition, this optimization also regularizes the joint state encoder, promoting balanced dependence on all views including the previous state-action pair.

We define Multi-View State Space Model (MV-SSM) with the following components obtained from the objective

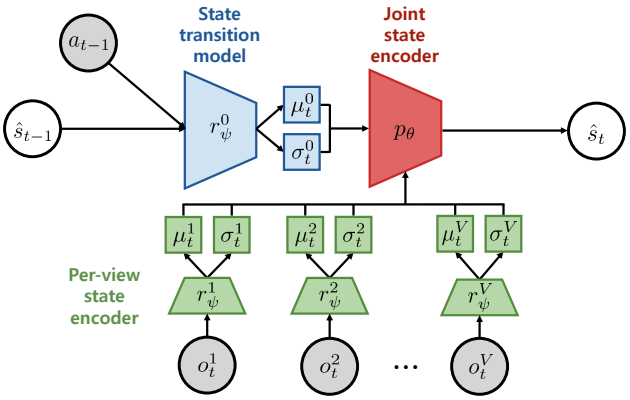


Figure 2. The state encoding process in F2C.

Eq. (10):

1. State transition model: $r^0_\psi(\hat{s}_t \mid \hat{s}_{t-1}, a_{t-1})$
2. Per-view observation model: $q^v_\phi(o^v_t \mid \hat{s}_t)$
3. Joint state encoder: $p_\theta(\hat{s}_t \mid \vec{o}_t, \hat{s}_{t-1}, a_{t-1})$
4. Per-view state encoder: $r^v_\psi(\hat{s}_t \mid o^v_t)$

4.4. Fuse2Control

We still need to address how to combine the latent states inferred from per-view encoders when only a subset of views is available.

Finally, we introduce Fuse2Control (F2C), an information-theoretic MVRL framework that retains *linear* scalability to multiple views and flexibility to missing views.

To this end, we follow the structural choice for the joint encoders in MVAE (Wu & Goodman, 2018) and MVTCAE (Hwang et al., 2021). Specifically, we design each per-view state encoder to be diagonal Gaussian such that $r^v_\psi(\hat{s}_t|o^v_t) = \mathcal{N}(\mu^v_t, (\sigma^v_t)^2 I)$ and the joint state encoder to be their inverse-variance weighted (IVW) (Cochran & Carroll, 1953; Cochran, 1954) average, such that

$$p_\theta(\hat{s}_t \mid \vec{o}_t, \hat{s}_{t-1}, a_{t-1}) \triangleq N(\mu_t, \sigma_t^2 \mathbf{I}), \tag{11}$$
$$\text{where } \mu_t \triangleq \frac{\sum_{v=0}^{V} \mu^v_t/(\sigma^v_t)^2}{\sum_{v=0}^{V} 1/(\sigma^v_t)^2} \text{ and } \sigma_t^2 \triangleq \frac{1}{\sum_{v=0}^{V} 1/(\sigma^v_t)^2},$$

where μ^0_t and σ^0_t are outputs of the state transition model $r^0_\psi(\hat{s}_t \mid \hat{s}_{t-1}, a_{t-1})$. Commonly used for sensor fusion, Eq. (11) allows the prior on the successive state inferred from the previous state to be corrected by the posterior information observed from any subset of views available at the moment. As a result, it encodes the multi-view observations and the state-action pair without introducing additional parameters ($\theta = \{\psi^v\}_{v=0}^V$), with the computation cost that *linearly* scales with the number of input views. Furthermore, missing views can be naturally excluded in the computation by setting σ^v_t of corresponding views to ∞.

In addition to its computational efficiency, our design choice for the joint state encoders to be an IVW average of the per-view state encoders is particularly well-suited to our formulation with CVIBs. This is because the calibrated per-view encoders are specifically designed to express uncertainty regarding unobserved factors, as discussed in Section 4.3. Since each per-view encoder outputs a Gaussian latent state, the dimensions associated with factors that are not observed from a given view yield large predicted variance $(\sigma^v_t)^2$. As a result, by averaging the per-view latent states weighted by the inverse of their predicted variances, we achieve a joint state that effectively aggregates information across views.

Overall, we name our method Fuse2Control (F2C), an information-theoretic MVRL framework that admits *linear* scalability with the number of views and capability to handle missing views during training and inference. Figure 2 describes the belief encoding process in F2C.

Using F2C, we can train the policy with any RL algorithm on top of the latent state extracted from the MV-SSM. The MV-SSM can be pretrained in advance (Section 5.1& Section 5.2) or jointly trained with policy (Section 5.3).

5. Experiments

To evaluate the quality of the learned latent state in various multi-view RL scenarios with missing views, we employ 3 sets of environments, Bipedal Walker, Simulation of Urban Mobility (SUMO), and Metaworld. Our main objective here is to see if the latent state from the proposed method aggregates task-relevant information from different modalities of views and helps the policy be robust to missing views. Due to the space limit, we mainly focus on the challenging missing-view pattern with the varying availability of all views (Zhang et al., 2019; 2020; Hwang et al., 2021). Discussion on other missing-view scenarios with additional results can be found in Section E.2.2 and Section E.2.3.

5.1. Bipedal Walker

To see if the latent state pretrained by our method is effective in missing-view scenarios given many heterogeneous views, we evaluated our method in the Bipedal Walker environment from OpenAI gym (Brockman et al., 2016). Equipped with various sensors, the agent needs to navigate in an environment of flat terrain with small variations. By controlling 4 motors in the legs, the agent receives a high reward if it travels far away from its initial position while receiving a huge negative reward whenever it falls down.

Multi-view observations We generate 5-heterogeneous-view observations which are angular positions (5), angular velocities (5), horizontal & vertical velocities (2), binary indicators of ground contact in two legs (2), and LIDAR measurements (10), where the numbers in parentheses are dimensions of views. Learning the latent state from these

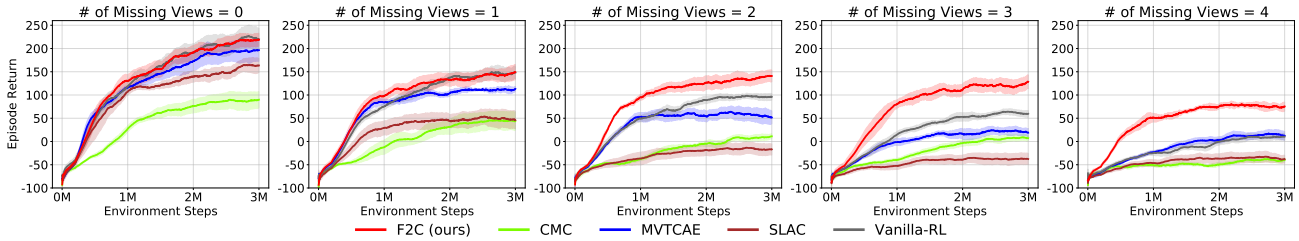


Figure 3. The performance in Bipedal Walker under missing views. From the left to the right, we increase the number of missing views.

multi-view observations is nontrivial since identifying the underlying relationship among them requires capturing temporal dependency. For example, positions depend on velocities not in the current timestep, but in the earlier timesteps.

Baseline methods We compared F2C to various baseline methods capable of learning from missing views such as CMC (Tian et al., 2020), SLAC (Lee et al., 2020a), MVTCAE (Hwang et al., 2021), and Vanilla-RL. CMC learns per-view representations by minimizing contrastive loss between every pair of views. Since CMC does not learn the joint representation, we compute the average of its representations from any available views. MVTCAE (Hwang et al., 2021) simultaneously learns joint representation and calibrates per-view representation encoders. SLAC jointly trains SSM and policy by maximizing the joint log-likelihood of observations and optimalities. Although it is a single-view RL method, SLAC can be extended to the MVRL by adopting the model structure same as MV-SSM in F2C and training it with Eq. (7). Lastly, Vanilla-RL is the PPO algorithm (Schulman et al., 2017) which learns directly from missing views. We used mean imputation (Van Buuren, 2018) in Vanilla-RL to cope with missing-view observations.

Experiment setup We first pretrained the representation of each method using a complete-view trajectory dataset. Based on the implementation in Bipedal Walker from Barhate (2021), the dataset is collected through interaction between the environment and PPO (Schulman et al., 2017) agent trained for 3 million timesteps. After pretraining all the methods, each method is evaluated by the performance of PPO trained on the learned representation. We adopted 5 different scenarios depending on how the representation is extracted from observations with a fixed number of missing views (0∼4). We uniformly sampled views to drop in each timestep. Details on the experiment can be found in Section D.1 in the supplementary material.

Results Figure 3 summarizes the performance of the policy trained with the representation learned by each method. All the values are averaged over 5 seeds (0∼4). All methods perform at their best when the observations are complete, but they all show a noticeable drop in their performances when 1 view is missing. This is mainly because the agent prefers a safe yet suboptimal behavior since any missing piece of information makes the walker significantly more challenging to keep running forward without falling down (see Section E.1.2 for visualization of exemplar behaviors).

Our method clearly outperforms all the baselines when there is more than 1 missing view. Vanilla-RL performed surprisingly well when there are only 0∼1 missing view, but significantly worse when more than one view is missing. We also observe that CMC poorly performs in all cases of complete and incomplete views. This is because the contrastive loss in CMC encourages the representations from all views to be similar, which enforces to discard of any view-specific information. Thus, it is limited to apply CMC when views are heterogeneous. Although our method also aligns per-view representations using CVIBs to the joint representation, it successfully aggregates all available information, which results in remarkable robustness to any number of missing views especially when 2 or more views are missing. We compare our method to MVTCAE and SLAC below.

Ablation study MVTCAE and SLAC are special cases of our F2C framework; F2C without relating latent states in adjacent timesteps reduces to MVTCAE (Eq. (3)) and F2C without per-view CVIBs reduces to SLAC (Eq. (7)). Compared to our method, MVTCAE and SLAC significantly degrade in all the missing-view cases. As shown in Figure 3, our method outperforms MVTCAE by a large margin when there are multiple missing views, which clearly shows the advantage of relating the current state to the previous state-action pair by learning the latent transition dynamics. Although SLAC also learns the transition dynamics, it fails to match its performance to F2C in all scenarios. This is because it does not regularize per-view encoders in its objective function, which forces the belief state encoder to merge inaccurate per-view representations. Further analysis can be found in Section E.1.

Runtime statistics The results of the training time of different representation learning methods are presented in Figure 4. The x-axis represents the number of input views, while the y-axis represents the computation time of a single iteration of the minibatch. We redundantly copied the original 5 views to employ many views. The numbers are averaged over 10 trials. Our method and SLAC have the same performance since the difference between them lies only in the coefficients of their objective function terms. The results clearly demonstrate the advantage of MVSSM

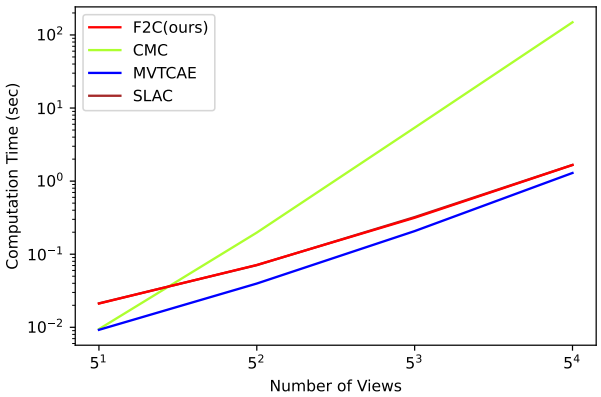


Figure 4. Average computation time of the optimization of each method per iteration. All the numbers are averaged over 10 trials.

(ours) over CMC. Although our method's computation time is slower than other methods when the number of views is 5, it gradually increases with the number of input views, similar to the MVTCAE whose computation cost also linearly scales with the number of views. In contrast, CMC shows a dramatic increase in computation time due to the optimization of the contrastive loss between every possible pair of views, which yields quadratic computation time.

5.2. Simulation of Urban Mobility

To see if our method can be effective in realistic scenarios, we evaluated our method in Simulation of Urban Mobility (SUMO) (Krajzewicz et al., 2012), a traffic light control environment. The goal is to manipulate traffic lights located at each junction to improve the overall traffic flow. We used the interface of 2×2 junctions provided by SUMO-RL (Alegre, 2019), where every junction has 4 vertical and 4 horizontal lanes. New vehicles are randomly generated with a probability of 0.1 at the end of lanes for every second.

Multi-view observations We take the observation from each junction as a view. In every junction, the following 21-dimensional information is observed to represent its state: (1) the current status of the traffic light represented by one-hot encoding (4D), (2) the duration time of the current traffic light status bounded between [0, 1] (1D), (3) the population density of all vehicles in each lane (8D), and (4) the population density of stopped vehicles in each lane (8D). Since all lanes in each junction are connected to other lanes in its adjacent junctions, there exists dependency across observations from all junctions. When some views are missing, learning the optimal behavior is challenging since the optimal action in each junction would be dependent on the observations from adjacent junctions which might be missing.

Experiment setup Same baseline methods and experiment setup in Section 5.1 are used except SAC (Haarnoja et al., 2018) is employed to train to policy. Further details can be found in Section D.2 in the supplementary material.

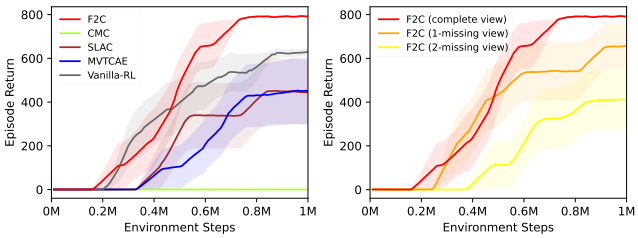


Figure 5. Experiment results in SUMO with complete views (left) and missing views (right).

Results The performances of ours and baseline methods are compared in Figure 5, where the left figure shows the results under complete view while the right figure shows the results under incomplete views. In the right figure, we only report the performance of ours and omit the baselines since they all yielded average returns of less than 5 when at least one view was missing.

Overall, F2C outperforms all the baseline methods in both complete-view and missing-view scenarios and performs reasonably robust to missing views. This result implies that our method discovers complex underlying dynamics across the traffic lights and the population densities. On the other hand, all the baseline methods fail to learn meaningful behavior when even one view is unavailable. Compared to Bipedal Walker, this is because the dependency across observations in SUMO is much weaker; the traffic conditions at adjacent junctions have stochastic effects on each other. Additional experiment results can be found in Section E.2.

5.3. Metaworld

Following Chen et al. (2021), we employed 8 complex robotic arm manipulation tasks in Metaworld (Yu et al., 2020) to see if our method accelerates the policy optimization when observations are high-dimensional.

Multi-view observations Three third-person-view cameras in different poses are used in each task to observe the robot arm agent and the objects. An example of 3-view image observations is visualized in Section D.3 in Appendix.

Baseline methods In addition to all the baseline methods in the previous experiments, we include LookCloser (Jangir et al., 2022), a transformer-based MVRL algorithm. A brief explanation of it can be found in Section 2. Due to the space limit, we compare the performance of our method for missing views. Evaluation in complete views with an extensive set of pixel-based RL algorithms can be found in Section E.3 in the supplementary material.

Experiment setup Following Chen et al. (2021), we evaluated the performance of the policies jointly trained with representations of comparing methods and ours to see if our method successfully captures task-relevant information in the latent state given high dimensional image observations. We applied the missing-view pattern randomly generated

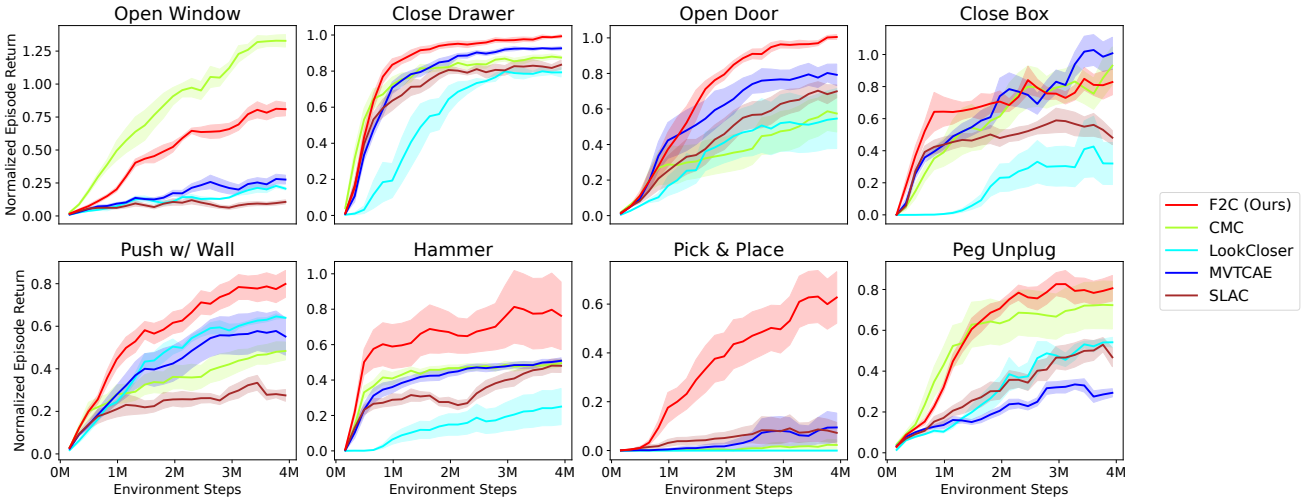


Figure 6. The performance in manipulation tasks given missing views ($\eta = 0.5$).

according to the protocol of Zhang et al. (2019). In specific, we generate random view-missing patterns per trajectory by setting the missing rate $\eta = \sum_{v=1}^{V} \frac{U_v}{(V \times T)}$ to be 0.5[1], where U_v is the number of samples missing in the v-th view. Detailed information on the experiment settings can be found in Section D.3 in the supplementary material.

Results Figure 6 summarizes the performance of F2C and comparing methods in the missing-view scenario. All values are normalized by the performance of F2C under complete views. Among 8 tasks, our method clearly outperforms in 5 tasks (`Close Drawer`, `Open Door`, `Push Wall`, `Hammer`, `Pick&Place`) and performs on par with the strongest baselines in 2 task (`Close Box`, `Peg Unplug`). Although our method underperforms in `Open Window` compared to CMC, we observe that our method shows stable performance across all tasks. For example, our method outperforms all the comparing methods in `Pick&Place`, while the rest algorithms barely learn any meaningful policies. Furthermore, we observe that the performances of F2C in complete views are reasonably well preserved in the missing-view experiment as well.

Unlike previous experiments, CMC shows competitive performance in some tasks (`Open Window`, `Close Box`, and `Peg Unplug`). This is because those 3 third-person-view cameras are redundant as they are commonly observing the same robot arms and objects. As a result, optimizing per-view representations with contrastive loss is less likely to discard any important information uniquely observable in some views. LookCloser fails to show competitive performance given missing-view observations, which implies that aggregating information across multi-view observations based on cross-attention is not effective in missing-view scenarios. Lastly, MVTCAE and SLAC show degenerate performance compared to ours, showing the impact of learning transition dynamics along with regularizing per-view encoders using CVIBs.

[1]We ensured that at least one view is always available while satisfying missing rate of 0.5.

6. Conclusion

We presented the latent space model for MVRL. Inspired by recent approaches in MVL, we derived the objective of (recurrent) state space models from TC and generalized it to multi-view settings. We also derived an alternative objective of MVRL with CVIBs, which is more favorable over naive log-likelihood objectives as it scales linearly with the number of views and able to handle missing views. To convey minimal sufficient latent state to policy, our method can be both pretrained with precollected dataset and trained end-to-end, achieving better performance compared to strong baseline methods in missing-view scenarios.

Acknowledgements

This work was supported by NRF of Korea (NRF2019R1A2C1087634 and NRF2021R1A4A3032834), Field-oriented Technology Development Project for Customs Administration through NRF of Korea funded by the MSIT and Korea Customs Service (NRF-2021M3I1A1097938), IITP grant funded by MSIT (No.2020-0-00940, Foundations of Safe Reinforcement Learning and Its Applications to Natural Language Processing; No.2022-0-00311, Development of Goal-Oriented Reinforcement Learning Techniques for Contact-Rich Robotic Manipulation of Everyday Objects; No.2019-0-00075, AI Graduate School Program (KAIST); No.2021-0-02068, AI Innovation Hub), ETRI grant (22ZS1100, Core Technology Research for Self-Improving Integrated AI System), KAIST-NAVER Hypercreative AI Center, and Samsung Electronics.

References

Alegre, L. N. SUMO-RL. https://github.com/LucasAlegre/sumo-rl, 2019.

Barhate, N. Minimal pytorch implementation of proximal policy optimization. https://github.com/nikhilbarhate99/PPO-PyTorch, 2021.

Belghazi, M. I., Baratin, A., Rajeshwar, S., Ozair, S., Bengio, Y., Courville, A., and Hjelm, D. Mutual information neural estimation. In *International Conference on Machine Learning*, 2018.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

Chen, B., Abbeel, P., and Pathak, D. Unsupervised learning of visual 3d keypoints for control. In *ICML*, 2021.

Cochran, W. G. The combination of estimates from different experiments. *Biometrics*, 1954.

Cochran, W. G. and Carroll, S. P. A sampling investigation of the efficiency of weighting inversely as the estimated variance. *Biometrics*, 1953.

Fan, J. and Li, W. DRIBO: Robust deep reinforcement learning via multi-view information bottleneck. In *International Conference on Machine Learning*, 2022.

Federici, M., Dutta, A., Forré, P., Kushman, N., and Akata, Z. Learning robust representations via multi-view information bottleneck. In *International Conference on Learning Representations*, 2020.

Fischer, I. The conditional entropy bottleneck, 2020.

Gao, S., Brekelmans, R., Ver Steeg, G., and Galstyan, A. Auto-encoding total correlation explanation. In *The 22nd International Conference on Artificial Intelligence and Statistics*, 2019.

Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, 2018.

Hafner, D., Lillicrap, T., Fischer, I., Villegas, R., Ha, D., Lee, H., and Davidson, J. Learning latent dynamics for planning from pixels. In *International Conference on Machine Learning*, 2019.

Hafner, D., Lillicrap, T., Ba, J., and Norouzi, M. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020.

Hafner, D., Lillicrap, T. P., Norouzi, M., and Ba, J. Mastering atari with discrete world models. In *International Conference on Learning Representations*, 2021.

Hjelm, R. D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., and Bengio, Y. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

Hu, Y., Sun, S., Xu, X., and Zhao, J. Attentive multi-view reinforcement learning. *International Journal of Machine Learning and Cybernetics*, 2020.

Hwang, H., Kim, G.-H., Hong, S., and Kim, K.-E. Variational interaction information maximization for cross-domain disentanglement. In *Advances in Neural Information Processing Systems*, 2020.

Hwang, H., Kim, G.-H., Hong, S., and Kim, K.-E. Multi-view representation learning via total correlation objective. *Advances in Neural Information Processing Systems*, 2021.

Jangir, R., Hansen, N., Ghosal, S., Jain, M., and Wang, X. Look closer: Bridging egocentric and third-person views with transformers for robotic manipulation. *IEEE Robotics and Automation Letters*, 2022.

Kaelbling, L. P., Littman, M. L., and Cassandra, A. R. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 1998.

Kaiser, L., Babaeizadeh, M., Milos, P., Osinski, B., Campbell, R. H., Czechowski, K., Erhan, D., Finn, C., Kozakowski, P., Levine, S., et al. Model-based reinforcement learning for atari. *arXiv preprint arXiv:1903.00374*, 2019.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Krajzewicz, D., Erdmann, J., Behrisch, M., and Bieker, L. Recent development and applications of sumo - simulation of urban mobility. *International Journal On Advances in Systems and Measurements*, December 2012.

Krishnan, R. G., Shalit, U., and Sontag, D. Deep kalman filters. *arXiv preprint arXiv:1511.05121*, 2015.

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., and Gershman, S. J. Building machines that learn and think like people. *Behavioral and brain sciences*, 2017.

Laskin, M., Lee, K., Stooke, A., Pinto, L., Abbeel, P., and Srinivas, A. Reinforcement learning with augmented data. *arXiv preprint arXiv:2004.14990*, 2020a.

Laskin, M., Srinivas, A., and Abbeel, P. Curl: Contrastive unsupervised representations for reinforcement learning. In *International Conference on Machine Learning*, 2020b.

Lee, A. X., Nagabandi, A., Abbeel, P., and Levine, S. Stochastic latent actor-critic: Deep reinforcement learning with a latent variable model. In *Advances in Neural Information Processing Systems*, 2020a.

Lee, J., Lee, B.-J., and Kim, K.-E. Reinforcement learning for control with multiple frequencies. In *Advances in Neural Information Processing Systems*, 2020b.

Li, L., Walsh, T. J., and Littman, M. L. Towards a unified theory of state abstraction for mdps. *ISAIM*, 2006.

Li, M., Wu, L., Ammar, H. B., and Wang, J. Multi-view reinforcement learning. In *Advances in Neural Information Processing Systems*, 2019.

McGill, W. Multivariate information transmission. *Transactions of the IRE Professional Group on Information Theory*, 1954.

Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Poklukar, P., Vasco, M., Yin, H., Melo, F. S., Paiva, A., and Kragic, D. Geometric multimodal contrastive representation learning. In *International Conference on Machine Learning*, 2022.

Poole, B., Ozair, S., Van Den Oord, A., Alemi, A., and Tucker, G. On variational bounds of mutual information. In *International Conference on Machine Learning*, 2019.

Rakelly, K., Gupta, A., Florensa, C., and Levine, S. Which mutual-information representation learning objectives are sufficient for control? In *Advances in Neural Information Processing Systems*, 2021.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Shi, Y., Siddharth, N., Paige, B., and Torr, P. H. Variational mixture-of-experts autoencoders for multi-modal deep generative models. *Advances in Neural Information Processing Systems*, 2019.

Sutter, T. M., Daunhawer, I., and Vogt, J. E. Multimodal generative learning utilizing jensen-shannon-divergence. *Advances in Neural Information Processing Systems*, 2020.

Sutter, T. M., Daunhawer, I., and Vogt, J. E. Generalized multimodal elbo. *International Conference on Learning Representations*, 2021.

Tassa, Y., Doron, Y., Muldal, A., Erez, T., Li, Y., Casas, D. d. L., Budden, D., Abdolmaleki, A., Merel, J., Lefrancq, A., et al. Deepmind control suite. *arXiv preprint arXiv:1801.00690*, 2018.

Tian, Y., Krishnan, D., and Isola, P. Contrastive multiview coding. In Vedaldi, A., Bischof, H., Brox, T., and Frahm, J.-M. (eds.), *European Conference on Computer Vision*, 2020.

Tishby, N., Pereira, F. C., and Bialek, W. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Van Buuren, S. *Flexible imputation of missing data*. CRC press, 2018.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

Ver Steeg, G. and Galstyan, A. Discovering structure in high-dimensional data through correlation explanation. In *Advances in Neural Information Processing Systems*, 2014.

Ver Steeg, G. and Galstyan, A. Maximally informative hierarchical representations of high-dimensional data. In *Artificial Intelligence and Statistics*, 2015.

Watanabe, S. Information theoretical analysis of multivariate correlation. *IBM Journal of research and development*, 1960.

Wu, M. and Goodman, N. Multimodal generative models for scalable weakly-supervised learning. *Advances in Neural Information Processing Systems*, 2018.

Yang, H., Shi, D., Xie, G., Peng, Y., Zhang, Y., Yang, Y., and Yang, S. Self-supervised representations for multi-view reinforcement learning. In *Conference on Uncertainty in Artificial Intelligence*, 2022.

Yu, T., Quillen, D., He, Z., Julian, R., Hausman, K., Finn, C., and Levine, S. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *International Conference on Robot Learning*, 2020.

Zhang, C., Han, Z., cui, y., Fu, H., Zhou, J. T., and Hu, Q. Cpm-nets: Cross partial multi-view networks. In *Advances in Neural Information Processing Systems*, 2019.

Zhang, C., Cui, Y., Han, Z., Zhou, J. T., Fu, H., and Hu, Q. Deep partial multi-view learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(5):2402–2415, 2020.

Supplementary Material

Contents

A. Q^*-Sufficiency Analysis on Our Objective

Data generation process Trajectories of observations and actions are collected by an unknown policy $\pi_D(a_t|\vec{o}_{\leq t})$ which is dependent on the history of observations with full support on action space $\mathcal{A}$. Then, trajectories are drawn from the following distribution.

$$p_D(a_{<T}, \vec{o}_{\leq T}) = \int p_D(s_{\leq T}, a_{<T}, \vec{o}_{\leq T}) ds_{\leq T}, \tag{12}$$

where $p_D(s_{\leq T}, a_{<T}, \vec{o}_{\leq T})$ is the unknown joint distribution of underlying ground-truth states, actions, and observations defined as below:

$$p_D(s_{\leq T}, a_{<T}, \vec{o}_{\leq T}) = p_0(s_0)\Omega(\vec{o}_0|s_0) \prod_{t=0}^{T-1} \pi_D(a_t|\vec{o}_{\leq t}) T\left(s_{t+1}|s_t, a_t\right) \Omega(\vec{o}_{t+1}|s_{t+1}).$$

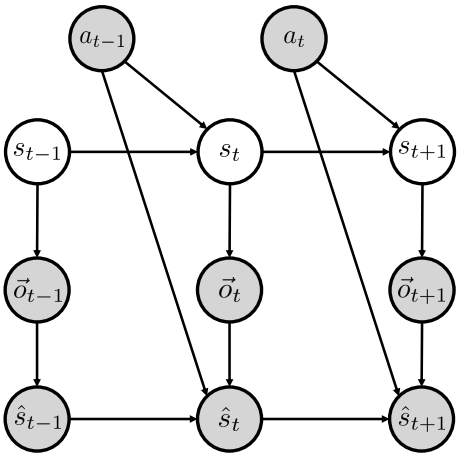


Figure 7. Graphical model of belief state encoding in MV-POMDP.

Although we do not have direct access to ground-truth states, we can also consider the joint distribution of the learned states along with ground-truth states, actions, and observations.

$$p_\theta(\hat{s}_{\leq T}, s_{\leq T}, a_{<T}, \vec{o}_{\leq T}) = p_D(s_{\leq T}, a_{<T}, \vec{o}_{\leq T}) p_\theta(\hat{s}_0|s_o) \textstyle\prod_{t=1}^{T} p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1}). \tag{13}$$

State-encoding distribution with full support Note that steady-state distribution is:

$$\mu_{\pi_D}(\mathrm{s}) = \lim_{t\to\infty} p_D(s_t = \mathrm{s}) = \lim_{t\to\infty} \int p_D(s_t = \mathrm{s}, s_{<t}, a_{<t}, \vec{o}_{\leq t}) ds_{<t} da_{<t} d\vec{o}_{\leq t}.$$

Since we assumed that $\mu_{\pi_D}(\mathrm{s}) > 0 \quad \forall \mathrm{s} \in \mathcal{S}$ (*i.e.* π_D is ergodic and thus μ_{π_D} have full support), we can extract the latent state $\hat{s}$ from the steady state-encoding distribution $\hat{\mu}_{\pi_D}(\mathrm{s})$ as below, whose support on the input s is the entire state space:

$$\hat{\mu}_{\pi_D}(\mathrm{s}) \triangleq \lim_{t\to\infty} p_\theta(\hat{s}_t|s_t = \mathrm{s}), \quad \text{where} \quad p_\theta(\hat{s}_t|s_t) = \frac{p_\theta(\hat{s}_t, s_t)}{p_D(s_t)} = \frac{\int p_\theta(\hat{s}_{\leq T}, s_{\leq T}, a_{<T}, \vec{o}_{\leq T}) \quad d\hat{s}_{<t} ds_{<t} da_{<t} d\vec{o}_{\leq t}}{\int p_D(s_{\leq T}, a_{<T}, \vec{o}_{\leq T}) \quad ds_{<t} da_{<t} d\vec{o}_{\leq t}}.$$

Q^*-sufficiency Let the optimal policy $\pi^* = \arg\max_{\pi\in\Pi} \mathbb{E}_\pi[\sum_{i=1}^{\infty} \gamma^{i-1} r(s_i)]$ and optimal Q-function $Q^* = Q^{\pi^*}$, where Π is the set of stationary policies. Please note that π^* is different from π_D; π^* is an optimal policy with direct access to the ground-truth state while π_D always covers full supports of the action space using the history of observations.

Following is the formal definition on Q^*-sufficiency (Li et al., 2006; Rakelly et al., 2021):
Definition 1 (Q^*-sufficiency). If $\hat{\mu}_{\pi_D}(\mathrm{s}^{(1)}) = \hat{\mu}_{\pi_D}(\mathrm{s}^{(2)})$, then $Q^*(\mathrm{s}^{(1)}, \mathrm{a}) = Q^*(\mathrm{s}^{(2)}, \mathrm{a}) \quad \forall \mathrm{a} \in \mathcal{A}, \forall r(s) \in \mathcal{R}$.

In Section A.2, we prove Theorem 1 by showing that if the belief state encoder maximizes $\forall t > 0\, TC_\theta(\langle \hat{S}_{t\text{-}1}, A_{t\text{-}1}\rangle, \vec{O}_t; \hat{S}_t)$, the latent state $\hat{s}$ encoded by $\boldsymbol{\theta}(s)$ is Q^*-sufficient w.r.t. a set of all state-dependent reward functions $\mathcal{R}$. Before we directly jump into the proof, we show in Section A.1 that following 3 lemmas hold, which are necessary for the proof:

A.1. Background Lemmas

Lemma 1. $I_\theta(\vec{O}_{t+1}; \hat{S}_t, A_t) \leq I_\theta(S_{t+1}; \hat{S}_t, A_t) \leq I(S_{t+1}; S_t, A_t), \quad \forall t \geq 0.$

Proof of Lemma 1.
(1) $I_\theta(\vec{O}_{t+1}; \hat{S}_t, A_t) \leq I_\theta(S_{t+1}; \hat{S}_t, A_t)$:

$$\begin{aligned} I_\theta(\hat{S}_t, A_t; S_{t+1}, \vec{O}_{t+1}) &= I_\theta\left(\hat{S}_t, A_t; S_{t+1}\right) + \cancelto{0}{I_\theta(\hat{S}_t, A_t; \vec{O}_{t+1} \mid S_{t+1})} \\ &= I_\theta(\hat{S}_t, A_t; \vec{O}_{t+1}) + I_\theta(\hat{S}_t, A_t; S_{t+1} \mid \vec{O}_{t+1}) \\ &\geq I_\theta(\hat{S}_t, A_t; \vec{O}_{t+1}) \end{aligned}$$

(2) $I_\theta(S_{t+1}; \hat{S}_t, A_t) \leq I\left(S_{t+1}; S_t, A_t\right)$:

$$\begin{aligned} I_\theta(S_{t+1}; S_t, \hat{S}_t, A_t) &= I(S_{t+1}; S_t, A_t) + \cancelto{0}{I_\theta(S_{t+1}; \hat{S}_t \mid S_t, A_t)} \\ &= I_\theta(S_{t+1}; \hat{S}_t, A_t) + I_\theta(S_{t+1}; S_t \mid \hat{S}_t, A_t) \\ &\geq I_\theta(S_{t+1}; \hat{S}_t, A_t) \end{aligned}$$

Thus, $I_\theta(\vec{O}_{t+1}; \hat{S}_t, A_t) \leq I(S_{t+1}; S_t, A_t) \quad \forall t \geq 0$. The equality holds when $\langle \hat{S}_t, A_t \rangle$ completely predicts $\vec{O}_{t+1}$. In such case, $I_\theta(\vec{O}_{t+1}; \hat{S}_t, A_t) = I_\theta(S_{t+1}; \hat{S}_t, A_t) = I(S_{t+1}; S_t, A_t)$. □

Lemma 2. $TC_\theta(\langle \hat{S}_t, A_t \rangle, \vec{O}_{t+1}; \hat{S}_{t+1}) \leq I_\theta(\hat{S}_t, A_t; \vec{O}_{t+1}) + TC(\vec{O}_{t+1}) \leq I(S_{t+1}; S_t, A_t) + TC(\vec{O}_{t+1}), \quad \forall t \geq 0.$

Proof of Lemma 2. To make the analysis easier, we shift $TC_\theta(\langle \hat{S}_{t-1}, A_{t-1} \rangle, \vec{O}_t; \hat{S}_t)$ one timestep ahead and rewrite it as follows:

$$TC_\theta(\langle \hat{S}_t, A_t \rangle, \vec{O}_{t+1}; \hat{S}_{t+1}) = TC_\theta(\langle \hat{S}_t, A_t \rangle, \vec{O}_{t+1}) - TC_\theta(\langle \hat{S}_t, A_t \rangle, \vec{O}_{t+1} \mid \hat{S}_{t+1}), \quad \text{where}$$

$$\begin{aligned} TC_\theta(\langle \hat{S}_t, A_t \rangle, \vec{O}_{t+1}) &= \mathbb{E}_{p_\theta(\hat{s}_t, a_t, \vec{o}_{t+1})}\left[\log \frac{p_\theta(\hat{s}_t, a_t, \vec{o}_{t+1})}{p_\theta(\hat{s}_t, a_t) \prod_{v=1}^V p_D(o_{t+1}^v)}\right] \\ &= \mathbb{E}_{p_\theta(\hat{s}_t, a_t, \vec{o}_{t+1})}\left[\log \frac{p_\theta\left(\hat{s}_t, a_t, \vec{o_{t+1}}\right)}{p_\theta\left(\hat{s}_t, a_t\right) p_D\left(\vec{o_{t+1}}\right)} \frac{p_D\left(\vec{o_{t+1}}\right)}{\prod_{v=1}^V p_D\left(o_{t+1}^v\right)}\right] \\ &= I_\theta(\hat{S}_t, A_t; \vec{O}_{t+1}) + TC(\vec{O}_{t+1}) \\ TC_\theta(\langle \hat{S}_t, A_t \rangle, \vec{O}_{t+1} \mid \hat{S}_{t+1}) &= \mathbb{E}_{p_\theta(\hat{s}_{t+1}, \hat{s}_t, a_t, \vec{o}_{t+1})}\left[\log \frac{p_\theta(\hat{s}_t, a_t, \vec{o}_{t+1} | \hat{s}_{t+1})}{p_\theta(\hat{s}_t, a_t | \hat{s}_{t+1}) \prod_{v=1}^V p_\theta(o_{t+1}^v | \hat{s}_{t+1})}\right] \\ &= \mathbb{E}_{p_\theta(\hat{s}_{t+1}, \hat{s}_t, a_t, \vec{o}_{t+1})}\left[\log \frac{p_\theta(\hat{s}_t, a_t, \vec{o}_{t+1} | \hat{s}_{t+1})}{p_\theta(\hat{s}_t, a_t | s_{t+1}) p_\theta(\vec{o}_{t+1} | \hat{s}_{t+1})} \frac{p_\theta(\vec{o}_{t+1} | \hat{s}_{t+1})}{\prod_{v=1}^V p_\theta(o_{t+1}^v | \hat{s}_{t+1})}\right] \\ &= I_\theta(\hat{S}_t, A_t; \vec{O}_{t+1} \mid \hat{S}_{t+1}) + TC_\theta(\vec{O}_{t+1} \mid \hat{S}_{t+1}). \end{aligned}$$

Thus, we obtain the following equality and inequalities:

$$\begin{aligned} TC_\theta(\langle \hat{S}_t, A_t \rangle, \vec{O}_{t+1}; \hat{S}_{t+1}) &= I_\theta(\hat{S}_t, A_t; \vec{O}_{t+1}) + TC(\vec{O}_{t+1}) \\ &\quad - I_\theta(\hat{S}_t, A_t; \vec{O}_{t+1} \mid \hat{S}_{t+1}) - TC_\theta(\vec{O}_{t+1} \mid \hat{S}_{t+1}) \qquad (14) \\ &\leq I_\theta(\hat{S}_t, A_t; \vec{O}_{t+1}) + TC(\vec{O}_{t+1}) \qquad (15) \\ &\leq I(S_t, A_t; S_{t+1}) + TC(\vec{O}_{t+1}), \qquad (16) \end{aligned}$$

where the inequality in Eq. (15) holds due to the nonnegativity of conditional MI & conditional TC and the inequality in Eq. (16) holds due to Lemma 1. Thus, our objective is bounded by $I(S_{t+1}; S_t, A_t) + TC(\vec{O}_{t+1})$, a constant value determined by the data distribution p_D.

The equality $TC_\theta(\langle \hat{S}_t, A_t \rangle, \vec{O}_{t+1}; \hat{S}_{t+1}) = I(S_t, A_t; S_{t+1}) + TC(\vec{O}_{t+1})$ holds if $\hat{S}_{t+1}$ minimizes the conditional MI and conditional TC in Eq. (14) (*i.e.,* complete representation of $\vec{O}_{t+1}$) in addition to the maximal predictive power of $\langle \hat{S}_t, A_t \rangle$ on $\vec{O}_{t+1}$. Thus, θ at the global optimum of our objective function ensures the equality $I_\theta(\hat{S}_t, A_t; \vec{O}_{t+1}) = I(S_t, A_t; S_{t+1})$. □

Lemma 3. For any $t \geq 0$, if $I_\theta(\vec{O}_{t+1}; \hat{S}_t, A_t) = I(S_{t+1}; S_t, A_t)$, then $T(s_{t+1}|s_t, a_t) = \mathbb{E}_{p_\theta(\hat{s}_t|s_t)}[p_\theta(s_{t+1}|\hat{s}_t, a_t)]$, $\forall s_t \in \operatorname{supp} p_t(\cdot), \forall \hat{s}_t \in \operatorname{supp} p_\theta(\cdot|s_t), \forall a_t$.

Proof of Lemma 3. By Lemma 1, $I_\theta(\vec{O}_{t+1}; \hat{S}_t, A_t) = I(S_{t+1}; S_t, A_t)$ directly indicates that $I_\theta(S_{t+1}; \hat{S}_t, A_t) = I(S_{t+1}; S_t, A_t)$. From Figure 7, we can derive the following equality:

$$\begin{aligned} I_\theta(S_{t+1}; \hat{S}_t, S_t, A_t) &= I(S_{t+1}; S_t, A_t) + \cancelto{0}{I_\theta(S_{t+1}; \hat{S}_t \mid S_t, A_t)} \\ &= I_\theta(S_{t+1}; \hat{S}_t, A_t) + I_\theta(S_{t+1}; S_t \mid \hat{S}_t, A_t). \end{aligned}$$

From $I_\theta(S_{t+1}; \hat{S}_t, A_t) = I(S_{t+1}; S_t, A_t)$,

$$\begin{aligned} I_\theta(S_{t+1}; S_t \mid \hat{S}_t, A_t) = 0 &= \mathbb{E}_{p_\theta(\hat{s}_t, s_t, a_t)}\left[D_{KL}\left[p_\theta(s_{t+1}|\hat{s}_t, s_t, a_t) \| p_\theta(s_{t+1}|\hat{s}_t, a_t)\right]\right] \\ &= \mathbb{E}_{p_\theta(a_t|\hat{s}_t, s_t) p_\theta(\hat{s}_t|s_t) p_D(s_t)}\left[D_{KL}\left[T(s_{t+1}|s_t, a_t) \| p_\theta(s_{t+1}|\hat{s}_t, a_t)\right]\right], \end{aligned}$$

where the last equality holds because S_{t+1} is conditionally independent of $\hat{S}_t$ given S_t, A_t ($I_\theta(S_{t+1}; \hat{S}_t \mid S_t, A_t) = 0$) so that $p_\theta(s_{t+1}|\hat{s}_t, s_t, a_t) = T(s_{t+1}|s_t, a_t)$.

If $p_D(s_t) > 0$ and $\hat{s}_t$ is the encoding of s_t so that $p_\theta(\hat{s}_t|s_t) > 0$, $p_\theta(a_t|s_t, \hat{s}_t)$ also covers all the supports of the action space as well since $p_\theta(\vec{o}_{\leq t}|s_t, \hat{s}_t)$ is well-defined:

$$p_\theta(a_t|s_t, \hat{s}_t) = \int \pi(a_t|\vec{o}_{\leq t}) p_\theta(\vec{o}_{\leq t}|s_t, \hat{s}_t) d\vec{o}_{\leq t} = \int \pi(a_t|\vec{o}_{\leq t}) \frac{p_\theta(\vec{o}_{\leq t}, s_t, \hat{s}_t)}{p_\theta(\hat{s}_t|s_t) p_D(s_t)} d\vec{o}_{\leq t}.$$

Thus, $\forall s_t \in \operatorname{supp} p_D(\cdot)$, $\forall \hat{s}_t \in \operatorname{supp} p_\theta(\cdot|s_t)$, and $\forall a_t$,

$$\begin{aligned} & T(s_{t+1}|s_t, a_t) = p_\theta(s_{t+1}|\hat{s}_t, a_t) \\ & \rightarrow T(s_{t+1}|s_t, a_t) p_\theta(\hat{s}_t|s_t) = p_\theta(s_{t+1}|\hat{s}_t, a_t) p_\theta(\hat{s}_t|s_t) \\ & \rightarrow T(s_{t+1}|s_t, a_t) = \mathbb{E}_{p_\theta(\hat{s}_t|s_t)}\left[p_\theta(s_{t+1}|\hat{s}_t, a_t)\right]. \end{aligned}$$

□

A.2. Q^*-Sufficiency of $TC_\theta(\langle \hat{S}_{t-1}, A_{t-1} \rangle, \vec{O}_t; \hat{S}_t)$

Given θ at the global optimum of $TC_\theta(\langle \hat{S}_{t-1}, A_{t-1} \rangle, \vec{O}_t; \hat{S}_t)$, $I_\theta(\hat{S}_t, A_t; \vec{O}_{t+1}) = I(S_t, A_t; S_{t+1}) \quad \forall t \geq 0$ by Lemma 2.

By Lemma 3 and stationary transition dynamics T in $\mathcal{M}$ such that $T(s_{t+1}|s_t = \mathrm{s}, a_t = \mathrm{a}) \equiv T(s_{t'+1}|s_{t'} = \mathrm{s}, a_{t'} = \mathrm{a})$ $\forall \mathrm{s}, \mathrm{a} \in \mathcal{S} \times \mathcal{A}, \forall t, t' \geq 0$, we have
$T(s_1|s_0 = \mathrm{s}, a_0 = \mathrm{a}) \equiv \lim_{t' \to \infty} \mathbb{E}_{p_\theta(\hat{s}_{t'}|s_{t'} = \mathrm{s})}\left[p_\theta(s_{t'+1}|\hat{s}_{t'}, a_{t'} = \mathrm{a})\right] = \lim_{t' \to \infty} \mathbb{E}_{\hat{\mathrm{s}} \sim \hat{\mu}_{\pi_D}(\mathrm{s})}\left[p_\theta(s_{t'+1}|\hat{s}_{t'} = \hat{\mathrm{s}}, a_{t'} = \mathrm{a})\right]$.

Based on this property, we prove Q^*-sufficiency using Lemma 3 as below.

Proof of Theorem 1. If two states $\mathrm{s}^{(1)}, \mathrm{s}^{(2)} \in \mathcal{S}$ are mapped into the same latent state space such that $\hat{\mu}_{\pi_D}(\mathrm{s}^{(1)}) = \hat{\mu}_{\pi_D}(\mathrm{s}^{(2)})$, their Q^* values are identical across all actions $\mathrm{a} \in \mathcal{A}$:

$$\begin{aligned} Q^*_r(\mathrm{s}^{(1)}, \mathrm{a}) &= \mathbb{E}_{\pi^*}\left[\sum_{i=1}^{\infty} \gamma^{i-1} R_i \mid s_0 = \mathrm{s}^{(1)}, a_0 = \mathrm{a}\right] \\ &= \mathbb{E}_{T(s_1|s_0=\mathrm{s}^{(1)}, a_0=\mathrm{a}) p^*(a_1, \ldots, s_\infty|s_1)}\left[\sum_{i=1}^{\infty} \gamma^{i-1} r(s_i)\right] \\ &= \lim_{t' \to \infty} \mathbb{E}_{\hat{\mathrm{s}} \sim \boldsymbol{\theta}(\mathrm{s}^{(1)})} \mathbb{E}_{p_\theta(s_{t'+1}|\hat{s}_{t'}=\hat{\mathrm{s}}, a_{t'}=\mathrm{a}) p^*(a_1, \ldots, s_\infty|s_1=s_{t'+1})}\left[\sum_{i=1}^{\infty} \gamma^{i-1} r(s_i)\right] \qquad (17) \\ &= \lim_{t' \to \infty} \mathbb{E}_{\hat{\mathrm{s}} \sim \boldsymbol{\theta}(\mathrm{s}^{(2)})} \mathbb{E}_{p_\theta(s_{t'+1}|\hat{s}_{t'}=\hat{\mathrm{s}}, a_{t'}=\mathrm{a}) p^*(a_1, \ldots, s_\infty|s_1=s_{t'+1})}\left[\sum_{i=1}^{\infty} \gamma^{i-1} r(s_i)\right] = Q^*_r(\mathrm{s}^{(2)}, \mathrm{a}), \end{aligned}$$

where the equality in Eq. (17) holds due to Lemma 3 and stationary transition dynamics.

Previous work has observed that any Q^*-sufficient representation guarantees the convergence of the Q-learning algorithm trained on top of the representation (Li et al., 2006; Rakelly et al., 2021). □

B. Derivations in Detail

Please note that all the derivations in this section relies on the joint distribution of observations, latent states, and actions, which is defined as:

$$p_\theta(\vec{o}_{\leq t}, \hat{s}_{\leq t}, a_{<t}) = p_D(\vec{o}_{\leq t}, a_{<t}) p_\theta(\hat{s}_0|\vec{o}_0) \prod_{t'=1}^{t} p_\theta(\hat{s}_{t'}|\vec{o}_{t'}, \hat{s}_{t'-1}, a_{t'-1}). \tag{18}$$

Please note that Eq. (18) is identical to marginalizing Eq. (13) over $s_{\leq t}$. Any joint distributions of some subsets of $\vec{o}_{\leq t}, \hat{s}_{\leq t}, a_{<t}$ can be achieved by marginalizing $p_\theta(\vec{o}_{\leq t}, \hat{s}_{\leq t}, a_{<t})$ w.r.t. random variables (RVs) that are absent in the given subset. For example,

$$p_\theta(\vec{o}_t, \hat{s}_t, \hat{s}_{t-1}, a_{t-1}) = \int p_\theta(\vec{o}_{\leq t}, \hat{s}_{\leq t}, a_{<t}) d\vec{o}_{<t} d\hat{s}_{<t-1} da_{<t-1},$$

which implies that sampling RVs at a certain timestep (or two consecutive timesteps) requires to sample RVs at earlier timesteps as well. In the example, samples of $\vec{o}_t, \hat{s}_t, \hat{s}_{t-1}, a_{t-1}$ can be achieved by sampling $\vec{o}_{\leq t}, \hat{s}_{\leq t}, a_{\leq t-1}$ and discarding $\vec{o}_{<t}, \hat{s}_{<t-1}, a_{<t-1}$. Samples of any subset of $\vec{o}_t, \hat{s}_t, \hat{s}_{t-1}, a_{t-1}$ can be achieved in similar ways.

B.1. Rewriting TC in terms of MI (Eq. (5))

Let O_t^0 denote $\langle \hat{S}_{t-1}, A_{t-1} \rangle$ to simplify notation. Then,

$$\begin{aligned}
TC_\theta(O_t^0, \vec{O}_t; \hat{S}_t) &= TC_\theta(O_t^0, \vec{O}_t) - TC_\theta(O_t^0, \vec{O}_t \mid \hat{S}_t) \\
&= D_{KL}\left(p_\theta(o^0, \vec{o}_t) \| p_\theta(o_t^0) \prod_{v=1}^{V} p_D(o_t^v) \right) - \mathbb{E}_{p_\theta(\hat{s}_t)} \left[D_{KL}\left(p_\theta(o_t^0, \vec{o}_t|\hat{s}_t) \| \prod_{v=0}^{V} p_\theta(o_t^v|\hat{s}_t) \right) \right] \\
&= H_\theta(O_t^0) + \sum_{v=1}^{V} H(O_t^v) - H_\theta(O_t^0, \vec{O}_t) - \sum_{v=0}^{V} H_\theta(O_t^v|\hat{S}_t) + H_\theta(O_t^0, \vec{O}_t|\hat{S}_t) \\
&= H_\theta(O_t^0) + \sum_{v=1}^{V} H(O_t^v) - \sum_{v=0}^{V} H_\theta(O_t^v|\hat{S}_t) - H_\theta(O_t^0, \vec{O}_t) + H_\theta(O_t^0, \vec{O}_t|\hat{S}_t) \\
&= \sum_{v=0}^{V} I_\theta(O_t^v; \hat{S}_t) - I_\theta(O_t^0, \vec{O}_t; \hat{S}_t)
\end{aligned} \tag{19}$$

B.2. Chain Rule for MI (Eq. (6) and Eq. (8))

In Eq. (6),

$$TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \vec{O}_t; \hat{S}_t) = I_\theta(\hat{S}_{t-1}, A_{t-1}; \hat{S}_t) + \sum_{v=1}^{V} I_\theta(O_t^v; \hat{S}_t) - I_\theta(\hat{S}_{t-1}, A_{t-1}, \vec{O}_t; \hat{S}_t)$$
$$= \sum_{v=1}^{V} I_\theta(O_t^v; \hat{S}_t) - I_\theta(\vec{O}_t; \hat{S}_t \mid \hat{S}_{t-1}, A_{t-1}),$$

where the first equality holds due to Eq. (19) and the last equality holds due to the chain rule for MI which we prove as below:

$$\begin{aligned}
&I_\theta(\hat{S}_{t-1}, A_{t-1}; \hat{S}_t) - I_\theta(\hat{S}_{t-1}, A_{t-1}, \vec{O}_t; \hat{S}_t) \\
&= \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t)}\left[\ln \frac{p_\theta(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})}{p_\theta(\hat{s}_t)}\right] - \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t, \vec{o}_t)}\left[\ln \frac{p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1})}{p_\theta(\hat{s}_t)}\right] \\
&= \int \left(\int p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t, \vec{o}_t) d\vec{o}_t\right) \ln \frac{p_\theta(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})}{p_\theta(\hat{s}_t)} d\hat{s}_{t-1} d\hat{a}_{t-1} d\hat{s}_t \\
&\quad - \int p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t, \vec{o}_t) \ln \frac{p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1})}{p_\theta(\hat{s}_t)} d\hat{s}_{t-1} da_{t-1} d\vec{o}_t d\hat{s}_t \\
&= \int p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t, \vec{o}_t)\left(\ln \frac{p_\theta(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})}{p_\theta(\hat{s}_t)} - \ln \frac{p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1})}{p_\theta(\hat{s}_t)}\right) d\hat{s}_{t-1} da_{t-1} d\vec{o}_t d\hat{s}_t \\
&= -\int p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t, \vec{o}_t) \ln \frac{p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1})}{p_\theta(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})} d\vec{o}_t d\hat{s}_t \\
&= -\mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t, \vec{o}_t)}\left[\ln \frac{p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1})}{p_\theta(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})}\right] \qquad = -I_\theta(\vec{O}_t; \hat{S}_t \mid \hat{S}_{t-1}, A_{t-1}). \quad (\star\star\star\star)
\end{aligned}$$

Similarly, in Eq. (8),

$$\begin{aligned}
TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \vec{O}_t; \hat{S}_t) &= \sum_{v=0}^{V} \tfrac{V+1}{V+1} I_\theta(O_t^v; \hat{S}_t) - \sum_{u=0}^{V} \tfrac{1}{V+1} I_\theta(\hat{S}_{t-1}, A_{t-1}, \vec{O}_t; \hat{S}_t) \\
&= \tfrac{1}{V+1} \sum_{v=0}^{V} \left[V \cdot I_\theta(O_t^v; \hat{S}) + I_\theta(O_t^v; \hat{S}_t) - I_\theta(\hat{S}_{t-1}, A_{t-1}, \vec{O}_t; \hat{S}_t)\right] \\
&= \tfrac{1}{V+1} \sum_{v=1}^{V} [V \cdot \underbrace{I_\theta(O_t^v; \hat{S}_t)}_{\star} - \underbrace{I_\theta(\hat{S}_{t-1}, A_{t-1}, \vec{O}_t^{\setminus v}; \hat{S}_t \mid O_t^v)}_{\star\star}] \\
&\quad + \tfrac{1}{V+1}\big(V \cdot \underbrace{I_\theta(\hat{S}_{t-1}, A_{t-1}; \hat{S}_t)}_{\star\star\star} - \underbrace{I_\theta(\vec{O}_t; \hat{S}_t \mid \hat{S}_{t-1}, A_{t-1})}_{\star\star\star\star}\big),
\end{aligned}$$

where the last equality holds due to the chain rule for MI which we prove as below:

$$\begin{aligned}
&I_\theta(O_t^v; \hat{S}_t) - I_\theta(\hat{S}_{t-1}, A_{t-1}, \vec{O}_t; \hat{S}_t) \\
&= \mathbb{E}_{p_\theta(\hat{s}_t, o_t^v)}\left[\ln \frac{p_\theta(\hat{s}_t|o_t^v)}{p_\theta(\hat{s}_t)}\right] - \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t, \vec{o}_t)}\left[\ln \frac{p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1})}{p_\theta(\hat{s}_t)}\right] \\
&= \int \left(\int p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t, \vec{o}_t) d\hat{s}_{t-1} da_{t-1} d\vec{o}_t^{\setminus v}\right) \ln \frac{p_\theta(\hat{s}_t|o_t^v)}{p_\theta(\hat{s}_t)} do_t^v d\hat{s}_t \\
&\quad - \int p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t, \vec{o}_t) \ln \frac{p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1})}{p_\theta(\hat{s}_t)} d\hat{s}_{t-1} da_{t-1} d\vec{o}_t d\hat{s}_t \\
&= \int p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t, \vec{o}_t)\left(\ln \frac{p_\theta(\hat{s}_t|o_t^v)}{p_\theta(\hat{s}_t)} - \ln \frac{p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1})}{p_\theta(\hat{s}_t)}\right) d\hat{s}_{t-1} da_{t-1} d\vec{o}_t d\hat{s}_t \\
&= -\int p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t, \vec{o}_t) \ln \frac{p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1})}{p_\theta(\hat{s}_t|o_t^v)} d\vec{o}_t d\hat{s}_t \\
&= -\mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t, \vec{o}_t)}\left[\ln \frac{p_\theta(\hat{s}_t|\hat{s}_{t-1}, a_{t-1}, \vec{o}_t^{\setminus v}, o_t^v)}{p_\theta(\hat{s}_t|o_t^v)}\right] \quad = -I_\theta(\hat{S}_{t-1}, A_{t-1}, \vec{O}_t^{\setminus v}; \hat{S}_t \mid O_t^v). \quad (\star\star)
\end{aligned}$$

B.3. Variational Lower Bound on $TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \vec{O}_t; \hat{S}_t)$ that Introduces a Transition Model (Eq. (7))

Variational lower bound Deriving a variational bound on each, we bypass the direct computation of MI terms in Eq. (6), $I_\theta(O_t^v; \hat{S}_t)$ and $I_\theta(\vec{O}_t; \hat{S}_t \mid \hat{S}_{t-1}, A_{t-1})$, which are involved with intractable integrals.

$$
\begin{aligned}
I_\theta(O_t^v; \hat{S}_t) &= \mathbb{E}_{p_\theta(\hat{s}_t, o_t^v)}\left[\ln \frac{p_\theta(o_t^v|\hat{s}_t)}{p_D(o_t^v)}\right] = \mathbb{E}_{p_\theta(\hat{s}_t, o_t^v)}\left[\ln \frac{q_\phi^v(o_t^v|\hat{s}_t) p_\theta(o_t^v|\hat{s}_t)}{p_D(o_t^v) q_\phi^v(o_t^v|\hat{s}_t)}\right] \\
&= H(O_t^v) + \mathbb{E}_{p_\theta(\hat{s}_t, o_t^v)}\left[\ln q_\phi^v(o_t^v|\hat{s}_t)\right] + \mathbb{E}_{p_\theta(\hat{s}_t)}\left[D_{KL}[p_\theta(o_t^v|\hat{s}_t)\|q_\phi^v(o_t^v|\hat{s}_t)]\right] \\
&\geq H(O_t^v) + \mathbb{E}_{p_\theta(\hat{s}_t, o_t^v)}\left[\ln q_\phi^v(o_t^v|\hat{s}_t)\right] \quad \text{and} \\
I_\theta(\vec{O}_t; \hat{S}_t \mid \hat{S}_{t-1}, A_{t-1}) &= \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t, \vec{o}_t)}\left[\ln \frac{p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1})}{p_\theta(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})}\right] \\
&= \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t, \vec{o}_t)}\left[\ln \frac{p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1}) r_\psi^0(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})}{r_\psi^0(\hat{s}_t|\hat{s}_{t-1}, a_{t-1}) p_\theta(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})}\right] \\
&= \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \vec{o}_t)}\left[D_{KL}[p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1})\|r_\psi^0(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})]\right] \\
&\quad - \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1})}\left[D_{KL}[p_\theta(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})\|r_\psi^0(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})]\right] \\
&\leq \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \vec{o}_t)}\left[D_{KL}[p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1})\|r_\psi^0(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})]\right].
\end{aligned}
$$

$$
\begin{aligned}
\text{Thus, } TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \vec{O}_t; \hat{S}_t) &= \sum_{v=1}^V I_\theta(O_t^v; \hat{S}_t) - I_\theta(\vec{O}_t; \hat{S}_t \mid \hat{S}_{t-1}, A_{t-1}) \\
&\geq \sum_{v=1}^V \left[H(O_t^v) + \mathbb{E}_{p_\theta(\hat{s}_t, o_t^v)}\left[\ln q_\phi^v(o_t^v|\hat{s}_t)\right]\right] \\
&\quad - \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \vec{o}_t)}\left[D_{KL}[p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1})\|r_\psi^0(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})]\right],
\end{aligned}
$$

which is identical to Eq. (7). It is important to note that the gap between $TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \vec{O}_t; \hat{S}_t)$ and Eq. (7) is as below:

$$
\sum_{v=1}^V \mathbb{E}_{p_\theta(\hat{s}_t)}\left[D_{KL}[p_\theta(o_t^v|\hat{s}_t)\|q_\phi^v(o_t^v|\hat{s}_t)]\right] + \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1})}\left[D_{KL}[p_\theta(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})\|r_\psi^0(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})]\right].
$$

Since $TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \vec{O}_t; \hat{S}_t)$ is upper bounded by a constant (see Lemma 2), maximization of Eq. (7) naturally fits $q_\phi^v(o_t^v|\hat{s}_t) \approx p_\theta(o_t^v|\hat{s}_t)$ and $r_\psi^0(\hat{s}_t|\hat{s}_{t-1}, a_{t-1}) \approx p_\theta(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})$.

Practical optimization As discussed at the beginning of Section B, sampling RVs at the certain timestep requires to sample RVs at earlier timesteps. Thus, we need to expand the expectations in Eq. (7) such that

$$\begin{aligned}\mathbb{E}_{p_\theta(\hat{s}_t, o_t^v)}\Big[\ln q_\phi^v(o_t^v|\hat{s}_t)\Big] &= \int \left(\int p_\theta(\vec{o}_{\leq t}, \hat{s}_{\leq t}, a_{<t}) d\vec{o}_{<t} d\vec{o}_t^{\setminus v} d\hat{s}_{<t} da_{<t}\right) \ln q_\phi^v(o_t^v|\hat{s}_t) do_t^v d\hat{s}_t \\ &= \int p_\theta(\vec{o}_{\leq t}, \hat{s}_{\leq t}, a_{<t}) \ln q_\phi^v(o_t^v|\hat{s}_t) d\vec{o}_{\leq t} d\hat{s}_{\leq t} da_{<t} = \mathbb{E}_{p_\theta(\vec{o}_{\leq t}, \hat{s}_{\leq t}, a_{<t})}\Big[\ln q_\phi^v(o_t^v|\hat{s}_t)\Big].\end{aligned}$$

$$\begin{aligned}&\mathbb{E}_{p_\theta(\hat{s}_{t\text{-}1}, a_{t\text{-}1}, \vec{o}_t)}\Big[D_{KL}[p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t\text{-}1}, a_{t\text{-}1}) \| r_\psi^0(\hat{s}_t|\hat{s}_{t\text{-}1}, a_{t\text{-}1})]\Big] \\ &= \int \left(\int p_\theta(\vec{o}_{\leq t}, \hat{s}_{<t}, a_{<t}) d\vec{o}_{<t} d\hat{s}_{<t\text{-}1} da_{<t\text{-}1}\right) D_{KL}[p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t\text{-}1}, a_{t\text{-}1}) \| r_\psi^0(\hat{s}_t|\hat{s}_{t\text{-}1}, a_{t\text{-}1})] d\hat{s}_{t\text{-}1} da_{t\text{-}1} d\vec{o}_t \\ &= \int p_\theta(\vec{o}_{\leq t}, \hat{s}_{<t}, a_{<t}) D_{KL}[p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t\text{-}1}, a_{t\text{-}1}) \| r_\psi^0(\hat{s}_t|\hat{s}_{t\text{-}1}, a_{t\text{-}1})] d\hat{s}_{<t} da_{<t} d\vec{o}_{\leq t} \\ &= \mathbb{E}_{p_\theta(\vec{o}_{\leq t}, \hat{s}_{<t}, a_{<t})}\Big[D_{KL}[p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t\text{-}1}, a_{t\text{-}1}) \| r_\psi^0(\hat{s}_t|\hat{s}_{t\text{-}1}, a_{t\text{-}1})]\Big].\end{aligned}$$

Thus, we used the objective function to train (MV-)SSM in SLAC (Lee et al., 2020a) (constant entropy terms are ignored).

$$\begin{aligned}&\text{Thus, } TC_\theta(\langle \hat{S}_{t\text{-}1}, A_{t\text{-}1}\rangle, \vec{O}_t; \hat{S}_t) \\ &\qquad \geq \sum_{v=1}^{V}\Big[H(O_t^v) + \mathbb{E}_{p_\theta(\vec{o}_{\leq t}, \hat{s}_{\leq t}, a_{<t})}\Big[\ln q_\phi^v(o_t^v|\hat{s}_t)\Big]\Big] \\ &\qquad\qquad - \mathbb{E}_{p_\theta(\vec{o}_{\leq t}, \hat{s}_{<t}, a_{<t})}\Big[D_{KL}[p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t\text{-}1}, a_{t\text{-}1}) \| r_\psi^0(\hat{s}_t|\hat{s}_{t\text{-}1}, a_{t\text{-}1})]\Big].\end{aligned}$$

B.4. Variational Lower Bound on $TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \vec{O}_t; \hat{S}_t)$ that Introduces Multiple CVIBs (Eq. (10))

Variational lower bound Similar to the previous section, we derive variational bounds on MI terms in Eq. (8)) to avoid intractability of them. Since variational bounds on $I_\theta(O_t^v; \hat{S}_t)$ ($\star$) and $I_\theta(\vec{O}_t; \hat{S}_t \mid \hat{S}_{t-1}, A_{t-1})$ ($\star\star\star\star$) are already derived in Section B.3, we provide a variational bounds on $I_\theta(\hat{S}_{t-1}, A_{t-1}, \vec{O}_t^{\setminus v}; \hat{S}_t \mid O_t^v)$ ($\star\star$). We refer the interested readers to Section 2.3 in (Poole et al., 2019) for the complete derivation of the variational lower bound on $I_\theta(\hat{S}_{t-1}, A_{t-1}; \hat{S}_t)$ ($\star\star\star$).

$$
\begin{aligned}
I_\theta(\hat{S}_{t-1}, A_{t-1}, \vec{O}_t^{\setminus v}; \hat{S}_t \mid O_t^v) &= \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t, \vec{o}_t)}\left[\ln \frac{p_\theta(\hat{s}_t|\hat{s}_{t-1}, a_{t-1}, \vec{o}_t^{\setminus v}, o_t^v)}{p_\theta(\hat{s}_t|o_t^v)}\right] \\
&= \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t, \vec{o}_t)}\left[\ln \frac{p_\theta(\hat{s}_t|\hat{s}_{t-1}, a_{t-1}, \vec{o}_t) r_\psi^v(\hat{s}_t|o_t^v)}{r_\psi^v(\hat{s}_t|o_t^v) p_\theta(\hat{s}_t|o_t^v)}\right] \\
&= \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \vec{o}_t)}\Big[D_{KL}[p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1}) \| r_\psi^v(\hat{s}_t|o_t^v)]\Big] \\
&\quad - \mathbb{E}_{p_D(o_t^v)}\Big[D_{KL}[p_\theta(\hat{s}_t|o_t^v) \| r_\psi^v(\hat{s}_t|o_t^v)]\Big] \\
&\le \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \vec{o}_t)}\Big[D_{KL}[p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1}) \| r_\psi^v(\hat{s}_t|o_t^v)]\Big]
\end{aligned}
$$

$$
\begin{aligned}
I_\theta(\hat{S}_{t-1}, A_{t-1}; \hat{S}_t) &= \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t)}\left[\ln \frac{p_\theta(\hat{s}_{t-1}, a_{t-1}, \hat{s}_t)}{p_\theta(\hat{s}_{t-1}, a_{t-1}) p_\theta(\hat{s}_t)}\right] \\
&\ge \mathbb{E}_{\prod_{k=1}^K p_\theta(\hat{s}_{t-1}^{(k)}, a_{t-1}^{(k)}, \hat{s}_t^{(k)})}\left[\frac{1}{K}\sum_{i=1}^K \log \frac{e^{g(\hat{s}_{t-1}^{(i)}, a_{t-1}^{(i)}, \hat{s}_t^{(i)})}}{\frac{1}{K}\sum_{j=1}^K e^{g(\hat{s}_{t-1}^{(j)}, a_{t-1}^{(j)}, \hat{s}_t^{(i)})}}\right] \triangleq I_{\text{NCE}}(t;\theta)
\end{aligned}
$$

$$
\begin{aligned}
TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \vec{O}_t; \hat{S}_t) &= \tfrac{1}{V+1}\sum_{v=1}^V [V \cdot \underbrace{I_\theta(O_t^v; \hat{S}_t)}_{\star} - \underbrace{I_\theta(\hat{S}_{t-1}, A_{t-1}, \vec{O}_t^{\setminus v}; \hat{S}_t \mid O_t^v)}_{\star\star}] \\
&\quad + \tfrac{1}{V+1}\big(V \cdot \underbrace{I_\theta(\hat{S}_{t-1}, A_{t-1}; \hat{S}_t)}_{\star\star\star} - \underbrace{I_\theta(\vec{O}_t; \hat{S}_t \mid \hat{S}_{t-1}, A_{t-1})}_{\star\star\star\star}\big), \\
&\ge \tfrac{V}{V+1} I_{\text{NCE}}(t;\theta) + \tfrac{V}{V+1}\sum_{v=1}^V \left[H(O_t^v) + \mathbb{E}_{p_\theta(o_t^v, \hat{s}_t)}\left[\ln q_\phi^v(o_t^v|\hat{s}_t)\right]\right] \\
&\quad - \tfrac{1}{V+1}\mathbb{E}\left[D_{KL}\left[p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1}) \| r_\psi^0(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})\right]\right] \\
&\quad - \tfrac{1}{V+1}\sum_{v=1}^V \mathbb{E}\left[D_{KL}\left[p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t-1}, a_{t-1}) \| r_\psi^v(\hat{s}_t|o_t^v)\right]\right] \\
&= TC_{\text{MV-SSM}}(t;\theta,\phi,\psi),
\end{aligned}
$$

which is identical to Eq. (10). When $I_{\text{NCE}}(t;\theta)$ tightly approximates $I_\theta(\hat{S}_{t-1}, A_{t-1}; \hat{S}_t)$ ($\star\star\star$), the gap between $TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \vec{O}_t; \hat{S}_t)$ and Eq. (10) is as below:

$$
\begin{aligned}
&\sum_{v=1}^V \left[\tfrac{V}{V+1} \cdot \mathbb{E}_{p_\theta(\hat{s}_t)}\Big[D_{KL}[p_\theta(o_t^v|\hat{s}_t) \| q_\phi^v(o_t^v|\hat{s}_t)]\Big] + \tfrac{1}{V+1} \cdot \mathbb{E}_{p_D(o_t^v)}\Big[D_{KL}[p_\theta(\hat{s}_t|o_t^v) \| r_\psi^v(\hat{s}_t|o_t^v)]\Big]\right] \\
&\quad + \tfrac{1}{V+1} \cdot \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1})}\Big[D_{KL}[p_\theta(\hat{s}_t|\hat{s}_{t-1}, a_{t-1}) \| r_\psi^0(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})]\Big].
\end{aligned}
$$

$TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \vec{O}_t; \hat{S}_t)$ is upper bounded by a constant as observed in Lemma 2. Thus, maximization of Eq. (10) naturally fits $r_\psi^v(\hat{s}_t|o_t^v) \approx p_\theta(\hat{s}_t|o_t^v)$ in addition to $q_\phi^v(o_t^v|\hat{s}_t) \approx p_\theta(o_t^v|\hat{s}_t)$ and $r_\psi^0(\hat{s}_t|\hat{s}_{t-1}, a_{t-1}) \approx p_\theta(\hat{s}_t|\hat{s}_{t-1}, a_{t-1})$.

Practical optimization As discussed at the beginning of Section B, sampling RVs at the certain timestep requires to sample RVs at earlier timesteps. Thus, we need to expand the expectations in Eq. (10) such that

$$
\begin{aligned}
&\mathbb{E}_{p_\theta(\hat{s}_{t-1},a_{t-1},\vec{o}_t)}\Big[D_{KL}[p_\theta(\hat{s}_t|\vec{o}_t,\hat{s}_{t-1},a_{t-1})\|r^v_\psi(\hat{s}_t|\vec{o}_t)]\Big] \\
&= \int\left(\int p_\theta(\vec{o}_{\le t},\hat{s}_{<t},a_{<t})d\vec{o}_{<t}d\hat{s}_{<t-1}da_{<t-1}\right)D_{KL}[p_\theta(\hat{s}_t|\vec{o}_t,\hat{s}_{t-1},a_{t-1})\|r^v_\psi(\hat{s}_t|\vec{o}_t)]d\hat{s}_{t-1}da_{t-1}d\vec{o}_t \\
&= \int p_\theta(\vec{o}_{\le t},\hat{s}_{<t},a_{<t})D_{KL}[p_\theta(\hat{s}_t|\vec{o}_t,\hat{s}_{t-1},a_{t-1})\|r^v_\psi(\hat{s}_t|\vec{o}_t)]d\hat{s}_{<t}da_{<t}d\vec{o}_{\le t} \\
&= \mathbb{E}_{p_\theta(\vec{o}_{\le t},\hat{s}_{<t},a_{<t})}\Big[D_{KL}[p_\theta(\hat{s}_t|\vec{o}_t,\hat{s}_{t-1},a_{t-1})\|r^v_\psi(\hat{s}_t|\vec{o}_t)]\Big].
\end{aligned}
$$

$$
\begin{aligned}
&\mathbb{E}_{\prod_{k=1}^K p_\theta(\hat{s}^{(k)}_{t-1},a^{(k)}_{t-1},\hat{s}^{(k)}_t)}\left[\frac{1}{K}\sum_{i=1}^K\log\frac{e^{g(\hat{s}^{(i)}_{t-1},a^{(i)}_{t-1},\hat{s}^{(i)}_t)}}{\frac{1}{K}\sum_{j=1}^K e^{g(\hat{s}^{(j)}_{t-1},a^{(j)}_{t-1},\hat{s}^{(i)}_t)}}\right] \\
&= \int\left(\int p_\theta(\vec{o}^{(1)}_{\le t},\hat{s}^{(1)}_{\le t},a^{(1)}_{<t})d\hat{s}^{(1)}_{<t-1}da^{(1)}_{<t-1}d\vec{o}^{(1)}_{\le t}\right)\prod_{k=2}^K p_\theta(\hat{s}^{(k)}_{t-1},a^{(k)}_{t-1},\hat{s}^{(k)}_t) \\
&\quad \frac{1}{K}\sum_{i=1}^K\log\frac{e^{g(\hat{s}^{(i)}_{t-1},a^{(i)}_{t-1},\hat{s}^{(i)}_t)}}{\frac{1}{K}\sum_{j=1}^K e^{g(\hat{s}^{(j)}_{t-1},a^{(j)}_{t-1},\hat{s}^{(i)}_t)}} \quad d\hat{s}^{(1)}_{t-1}da^{(1)}_{t-1}d\hat{s}^{(1)}_t \quad d\hat{s}^{(2)}_{t-1}da^{(2)}_{t-1}d\hat{s}^{(2)}_t...d\hat{s}^{(K)}_{t-1}da^{(K)}_{t-1}d\hat{s}^{(K)}_t \\
&= \int p_\theta(\vec{o}^{(1)}_{\le t},\hat{s}^{(1)}_{\le t},a^{(1)}_{<t})\prod_{k=2}^K p_\theta(\hat{s}^{(k)}_{t-1},a^{(k)}_{t-1},\hat{s}^{(k)}_t)\sum_{i=1}^K\log\frac{e^{g(\hat{s}^{(i)}_{t-1},a^{(i)}_{t-1},\hat{s}^{(i)}_t)}}{\frac{1}{K}\sum_{j=1}^K e^{g(\hat{s}^{(j)}_{t-1},a^{(j)}_{t-1},\hat{s}^{(i)}_t)}} \\
&\quad d\hat{s}^{(1)}_{\le t}da^{(1)}_{<t}d\vec{o}^{(1)}_{\le t} \quad d\hat{s}^{(2)}_{t-1}da^{(2)}_{t-1}d\hat{s}^{(2)}_t...d\hat{s}^{(K)}_{t-1}da^{(K)}_{t-1}d\hat{s}^{(K)}_t \\
&= \mathbb{E}_{p_\theta(\vec{o}^{(1)}_{\le t},\hat{s}^{(1)}_{\le t},a^{(1)}_{<t})\prod_{k=2}^K p_\theta(\hat{s}^{(k)}_{t-1},a^{(k)}_{t-1},\hat{s}^{(k)}_t)}\left[\frac{1}{K}\sum_{i=1}^K\log\frac{e^{g(\hat{s}^{(i)}_{t-1},a^{(i)}_{t-1},\hat{s}^{(i)}_t)}}{\frac{1}{K}\sum_{j=1}^K e^{g(\hat{s}^{(j)}_{t-1},a^{(j)}_{t-1},\hat{s}^{(i)}_t)}}\right].
\end{aligned}
$$

Repeating the same procedures for K times, we end up with

$$
I_{\text{NCE}}(t;\theta) = \mathbb{E}_{\prod_{k=1}^K p_\theta(\vec{o}^{(k)}_{\le t},\hat{s}^{(k)}_{\le t},a^{(k)}_{<t})}\left[\frac{1}{K}\sum_{i=1}^K\log\frac{e^{g(\hat{s}^{(i)}_{t-1},a^{(i)}_{t-1},\hat{s}^{(i)}_t)}}{\frac{1}{K}\sum_{j=1}^K e^{g(\hat{s}^{(j)}_{t-1},a^{(j)}_{t-1},\hat{s}^{(i)}_t)}}\right].
$$

Finally, following is the objective function we used to train MV-SSM in Fuse2Control (constant entropy terms are ignored).

$$
\begin{aligned}
TC_\theta(\langle\hat{S}_{t-1},A_{t-1}\rangle,\vec{O}_t;\hat{S}_t) \ge\ & \tfrac{V}{V+1}\mathbb{E}_{\prod_{k=1}^K p_\theta(\vec{o}^{(k)}_{\le t},\hat{s}^{(k)}_{\le t},a^{(k)}_{<t})}\left[\frac{1}{K}\sum_{i=1}^K\log\frac{e^{g(\hat{s}^{(i)}_{t-1},a^{(i)}_{t-1},\hat{s}^{(i)}_t)}}{\frac{1}{K}\sum_{j=1}^K e^{g(\hat{s}^{(j)}_{t-1},a^{(j)}_{t-1},\hat{s}^{(i)}_t)}}\right] \\
&+ \tfrac{V}{V+1}\sum_{v=1}^V\left[H(O^v_t)+\mathbb{E}_{p_\theta(\vec{o}_{\le t},\hat{s}_{\le t},a_{<t}))}\left[\ln q^v_\phi\left(o^v_t|\hat{s}_t\right)\right]\right] \\
&- \tfrac{1}{V+1}\mathbb{E}_{p_\theta(\vec{o}_{\le t},\hat{s}_{<t},a_{<t})}\left[D_{KL}\left[p_\theta\left(\hat{s}_t|\vec{o}_t,\hat{s}_{t-1},a_{t-1}\right)\|r^0_\psi\left(\hat{s}_t|\hat{s}_{t-1},a_{t-1}\right)\right]\right] \\
&- \tfrac{1}{V+1}\sum_{v=1}^V\mathbb{E}_{p_\theta(\vec{o}_{\le t},\hat{s}_{<t},a_{<t})}\left[D_{KL}\left[p_\theta\left(\hat{s}_t|\vec{o}_t,\hat{s}_{t-1},a_{t-1}\right)\|r^v_\psi\left(\hat{s}_t|o^v_t\right)\right]\right] \\
=\ & TC_{\text{MV-SSM}}(t;\theta,\phi,\psi),
\end{aligned}
$$

B.5. Investigation on $I_\theta(\hat{S}_{t\text{-}1}, A_{t\text{-}1}; \hat{S}_t)$

$$
\begin{aligned}
I_\theta(\hat{S}_{t\text{-}1}, A_{t\text{-}1}; \hat{S}_t) &= \mathbb{E}_{p_\theta(\hat{s}_t, \hat{s}_{t\text{-}1}, a_{t\text{-}1})}\left[\ln \frac{p_\theta(\hat{s}_{t\text{-}1}, a_{t\text{-}1}|\hat{s}_t)}{p_\theta(\hat{s}_{t\text{-}1}, a_{t\text{-}1})}\right] \\
&= \mathbb{E}_{p_\theta(\hat{s}_t, \hat{s}_{t\text{-}1}, a_{t\text{-}1})}\left[\ln \frac{p_\theta(\hat{s}_{t\text{-}1}, a_{t\text{-}1}|\hat{s}_t) q_\phi(\hat{s}_{t\text{-}1}, a_{t\text{-}1}|\hat{s}_t)}{p_\theta(\hat{s}_{t\text{-}1}, a_{t\text{-}1}) q_\phi(\hat{s}_{t\text{-}1}, a_{t\text{-}1}|\hat{s}_t)}\right] \\
&= H_\theta(\hat{s}_{t\text{-}1}, a_{t\text{-}1}) + \mathbb{E}_{p_\theta(\hat{s}_t, \hat{s}_{t\text{-}1}, a_{t\text{-}1})}[q_\phi(\hat{s}_{t\text{-}1}, a_{t\text{-}1}|\hat{s}_t)] \\
&\quad + \mathbb{E}_{p_\theta(\hat{s}_t)}\left[D_{KL}[p_\theta(\hat{s}_{t\text{-}1}, a_{t\text{-}1}|\hat{s}_t)||q_\phi(\hat{s}_{t\text{-}1}, a_{t\text{-}1}|\hat{s}_t)]\right] \\
&\geq H_\theta(\hat{S}_{t\text{-}1}, A_{t\text{-}1}) + \mathbb{E}_{p_\theta(\hat{s}_t, \hat{s}_{t\text{-}1}, a_{t\text{-}1})}[q_\phi(\hat{s}_{t\text{-}1}, a_{t\text{-}1}|\hat{s}_t)] \qquad (20)
\end{aligned}
$$

Unlike observations drawn from the data distribution $p_D(\vec{o}_{\leq T}, a_{\leq T})$ each of which has constant entropy $H(O^v)$, the first term $H_\theta(\hat{S}_{t\text{-}1}, A_{t\text{-}1})$ in Eq. (20) is no longer constant since it is a function of $\hat{s}_{t\text{-}1}$ encoded by the $p_\theta(\hat{s}_{t\text{-}1}|\vec{o}_{t\text{-}1}, \hat{s}_{t\text{-}2}, a_{t\text{-}2})$ as we discussed in Section 4.1. Thus, reconstruction of $\hat{s}_{t\text{-}1}, a_{t\text{-}1}$ from $\hat{s}_t$ does not necessarily maximize $I_\theta(\hat{S}_{t\text{-}1}, A_{t\text{-}1}; \hat{S}_t)$; maximization of the entropy term in addition to maximization of the second term (the negative reconstruction term) in Eq. (20) is required. Furthermore, maximizing the second term yields a trivial solution since both p_θ and q_ϕ are parameterized distributions where their outputs can easily be matched to meaningless values.

To resolve the issue, we can adopt sample-based MI estimators (Belghazi et al., 2018; Oord et al., 2018; Hjelm et al., 2018; Poole et al., 2019) to lower bound $I_\theta(\hat{S}_{t\text{-}1}, A_{t\text{-}1}; \hat{S}_t)$ since sampling $\hat{s}_t, \vec{o}_t, \hat{s}_{t\text{-}1}, a_{t\text{-}1}$ from the density $p_\theta\left(\hat{s}_t, \vec{o}_t, \hat{s}_{t\text{-}1}, a_{t\text{-}1}\right)$ is available without computing the density. Among these estimators which lower bound MI, we found that noise contrastive estimation (NCE) (Oord et al., 2018) shows numerical stability in the optimization due to its low variance estimation as observed by (Poole et al., 2019).

C. Connection to Previous Works

C.1. Connection to RSSM

Given only one observation from single view, setting $O_t^0 = (\hat{S}_{<t}, A_{<t})$ reduces to the model close to RSSM.

$$
\begin{aligned}
TC_\theta(O_t^0, \vec{O}_t; \hat{S}_t) &= \sum_{v=1}^{V} I_\theta(O_t^v; \hat{S}_t) + \underline{I_\theta(\hat{S}_{<t}, A_{<t}; \hat{S}_t)} - I_\theta(\hat{S}_{<t}, A_{<t}, \vec{O}_t; \hat{S}_t) \\
&= \sum_{v=1}^{V} I_\theta(O_t^v; \hat{S}_t) - I_\theta(\vec{O}_t; \hat{S}_t \mid \hat{S}_{<t}, A_{<t}) \\
&\geq \sum_{v=1}^{V} H(O_t^v) + \sum_{v=1}^{V} \mathbb{E}_{p_\theta(o_t^v, o_t)}[\ln q_\phi(o_t^v|\hat{s}_t)] \\
&\quad - \mathbb{E}_{p_\theta(\vec{o}_t, \hat{s}_{<t}, o_{<t})}\Big[D_{KL}\big(p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{<t}, a_{<t}) \| \underline{r_\psi(\hat{s}_t|\hat{s}_{<t}, a_{<t})}\big)\Big] \qquad (21)
\end{aligned}
$$

Plugging $V = 1$, Eq. (21) matches to RSSM (Hafner et al., 2019; 2020; 2021), only with minor difference in decoders. In the original RSSM, the decoder receives o_t^0 in addition to $\hat{s}_t$ while the decoder in Eq. (21) only receives $\hat{s}_t$.

Since Hafner et al. (2019; 2020; 2021) aim to optimize the ELBO of trajectories, their derivation decomposes ELBO for every timestep. As a result, the observation model (decoder) in their formulation is conditioned on the history of previous latent states and actions in addition to the current latent state. This makes training the observation model straightforward. However, the observation model could rely on the history instead of the current latent state, resulting in non-Markov latent states, which is explicitly addressed in our work.

C.2. Connection to MVTCAE

Due to nonnegativity of (conditional) TC, following inequality holds:

$$
\begin{aligned}
TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \vec{O}_t; \hat{S}_t) &= TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \vec{O}_t) - TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \vec{O}_t \mid \hat{S}_t), \quad \text{where} \\
TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \vec{O}_t) &= \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \vec{o}_t)}\left[\log \frac{p_\theta(\hat{s}_{t-1}, a_{t-1}, \vec{o}_t)}{p_\theta(\hat{s}_{t-1}, a_{t-1}) \prod_{v=1}^V p_D(o_t^v)}\right] \\
&= \mathbb{E}_{p_\theta(\hat{s}_{t-1}, a_{t-1}, \vec{o}_t)}\left[\log \frac{p_\theta\left(\hat{s}_{t-1}, a_{t-1}, \vec{o}_t\right)}{p_\theta\left(\hat{s}_{t-1}, a_{t-1}\right) p_D\left(\vec{o}_t\right)} \frac{p_D\left(\vec{o}_t\right)}{\prod_{v=1}^V p_D\left(o_t^v\right)}\right] \\
&= I_\theta(\hat{S}_{t-1}, A_{t-1}; \vec{O}_t) + TC(\vec{O}_t) \\
TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \vec{O}_t \mid \hat{S}_t) &= \mathbb{E}_{p_\theta(\hat{s}_t, \hat{s}_{t-1}, a_{t-1}, \vec{o}_t)}\left[\log \frac{p_\theta(\hat{s}_{t-1}, a_{t-1}, \vec{o}_t | \hat{s}_t)}{p_\theta(\hat{s}_{t-1}, a_{t-1} | \hat{s}_t) \prod_{v=1}^V p_\theta(o_t^v | \hat{s}_t)}\right] \\
&= \mathbb{E}_{p_\theta(\hat{s}_t, \hat{s}_{t-1}, a_{t-1}, \vec{o}_t)}\left[\log \frac{p_\theta(\hat{s}_{t-1}, a_{t-1}, \vec{o}_t | \hat{s}_t)}{p_\theta(\hat{s}_{t-1}, a_{t-1} | s_t) p_\theta(\vec{o}_t | \hat{s}_t)} \frac{p_\theta(\vec{o}_t | \hat{s}_t)}{\prod_{v=1}^V p_\theta(o_t^v | \hat{s}_t)}\right] \\
&= I_\theta(\hat{S}_{t-1}, A_{t-1}; \vec{O}_t \mid \hat{S}_t) + TC_\theta(\vec{O}_t \mid \hat{S}_t).
\end{aligned}
$$

Thus, our objective function can be rewritten as

$$
\begin{aligned}
TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \vec{O}_t; \hat{S}_t) &= I_\theta(\hat{S}_{t-1}, A_{t-1}; \vec{O}_t) + TC(\vec{O}_t) \\
&\quad - I_\theta(\hat{S}_{t-1}, A_{t-1}; \vec{O}_t \mid \hat{S}_t) - TC_\theta(\vec{O}_t \mid \hat{S}_t) \\
&= I_\theta(\hat{S}_{t-1}, A_{t-1}; \vec{O}_t) - I_\theta(\hat{S}_{t-1}, A_{t-1}; \vec{O}_t \mid \hat{S}_t) \qquad (22) \\
&\quad + TC(\vec{O}_t) - TC_\theta(\vec{O}_t \mid \hat{S}_t) \\
&= I_\theta(\hat{S}_{t-1}, A_{t-1}; \vec{O}_t; \hat{S}_t)) + TC_\theta(\vec{O}_t; \hat{S}_t) \qquad (23) \\
\text{(Alternative)} \quad &= TC_\theta(\langle \hat{S}_{t-1}, A_{t-1}\rangle, \langle \vec{O}_t\rangle; \hat{S}_t) + TC_\theta(\vec{O}_t; \hat{S}_t), \qquad (24)
\end{aligned}
$$

where the first term $I_\theta(\hat{S}_{t-1}, A_{t-1}; \vec{O}_t; \hat{S}_t)$ in Eq. (23) is known as Interaction Information (II) (McGill, 1954; Hwang et al., 2020), another generalization of MI which also measures dependency among multiple random variables similar to TC. Eq. (22) implies that $I_\theta(\hat{S}_{t-1}, A_{t-1}; \vec{O}_t; \hat{S}_t)$ is maximized when the dependency between the previous state-action pair and the current observation is maximized while the dependency of them is conditionally minimized given the current latent state. Thus, optimization of $I_\theta(\hat{S}_{t-1}, A_{t-1}; \vec{O}_t; \hat{S}_t)$ yields the Markov property in the space of the learned latent state. Treating $\langle \vec{O}_t \rangle$ as one joint random variable similar to $\langle \hat{S}_{t-1}, A_{t-1}\rangle$, II can be also written in TC as in Eq. (24).

Interestingly, the second term in Eq. (23) is the objective function of MVTCAE along with joint encoder $p_\theta(s_t | \vec{o}_t)$, which are also the objective function and the joint belief state encoder of MV-SSM in the first timestep $t = 0$. Thus, Eq. (23) and Eq. (24) clearly indicate that our objective function allows the latent state both to capture the temporal dependency to relate two consecutive latent states and to learn the complete representation of multi-view observations (Hwang et al., 2021).

D. Details of Experiment

We provide detailed information about the experimental setup and hyperparameter settings in each environment. To make fair comparisons in each environment, we followed the choice of policy optimization algorithm and its hyperparameters determined by the publisher of each code base; PPO implementation in Bipedal Walker from Barhate (2021), SAC implementation in SUMO from Lee et al. (2020b), and PPO implementation in Metaworld from Chen et al. (2021). In each environment, the same network architectures for encoders are applied to CMC, MVTCAE, SLAC, and F2C as well as the same decoders for MVTCAE, SLAC, and F2C, and the same dynamics models for SLAC and F2C. To train the dynamics models for SLAC and F2C, we employed 3 fully connected layers in Bipedal Walker and Metaworld, and 1 LSTM layer & 1 fully connected layer in SUMO. The transition dynamics model introduces one additional hyperparameter H which is the length of the history of the past observations and actions. In addition, we chose the size of the latent representations to be 24 and 84 dimensions in Bipedal Walker and SUMO respectively, which match the size of the stacked complete-view observations in these environments. This ensures that the policies of all methods have the same number of parameters as Vanilla-RL, which directly learns from the raw observation of all views. In Metaworld, the size of the representation is 128, which follows Keypoint3D (Chen et al., 2021). All networks in policies and representations models are optimized by Adam (Kingma & Ba, 2015) for all environments. The code is available at: `https://github.com/gr8joo/F2C`

Hyperparameter	Bipedal Walker	Metaworld
Number of Views (N)	5	3
Policy	PPO	PPO
PPO batch size	4,000	6,400
Rollout buffer size	4,000	100,000
# Epochs per update	80	8
Gamma	0.99	0.99
GAE lambda	-	0.95
Clip range (ϵ)	0.2	0.2
Entropy coefficient	0.005	0.0
Value function coefficient	-	0.5
Gradient clip	-	0.5
Target KL	-	0.12
Learning rate (actor)	3e-4	3e-4
Learning rate (critic)	1e-3	3e-4
Learning rate (representation)	3e-4	3e-4
Observation buffer size	-	100,000
# Unsupervised learning steps	-	400
Subsequence length (H)	8	4
Size of the latent state (representation)	24	128

Table 1. The summary of hyperparameters in Bipedal Walker and Metaworld.

D.1. Bipedal Walker

Bipedal Walker is an OpenAI gym (Brockman et al., 2016) environment built on the Box2D physics engine. Equipped with LIDAR sensors, the agent needs to navigate a flat terrain with small random variations. The agent's heavy hull is supported by two legs. By controlling 4 motors in the legs (2 in the hips and the other 2 in the knees), the agent receives high reward if it travels far away from its initial position while receiving large negative reward when it falls down.

Baseline methods CMC, MVTCAE, SLAC, and Vanilla-RL (PPO).

Experiment setup The following 3-step evaluation procedure is applied to all representation learning methods:

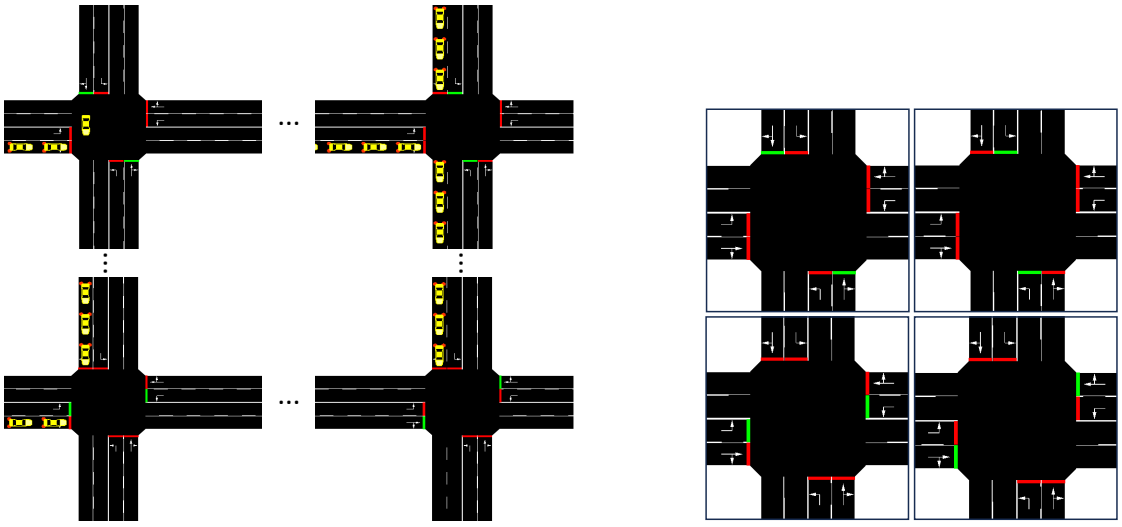
1. Collect a trajectory dataset composed of complete-view observations and actions by training a PPO policy from scratch for 3 million timesteps.
2. Pretrain each method across 5 seeds (0∼4) for 50 epochs using the precollected dataset.
3. For every fixed number of missing views (0∼4), train PPO for 3 million timesteps on top of the frozen latent state extracted from each pretrained method with its corresponding seed. All the accumulated rewards of each method is

averaged over 5 seeds (0∼4).

The dataset collected in Step 1 was split into train (0.8) and validation (0.2) sets, where the numbers in parentheses are approximate ratios. In step 2, we saved the weights of the model when the loss evaluated with validation set is at its minimum. Please note that Vanilla-RL directly trained PPO with multi-view observations from the environment filling the missing view observations with mean values of the train data. For the given fixed number of missing views, the views to be missing were uniformly and randomly chosen in every timestep. Detailed information on the hyperparameter settings can be found in Table 1.

D.2. SUMO

Simulation of Urban Mobility (SUMO) (Krajzewicz et al., 2012) is a realistic traffic light control environment. The goal is to manipulate traffic lights located at each junction to improve the overall traffic flow. Specifically, the agent in the environment needs to maximize the accumulative reward where the reward in every timestep is defined to be $\min_{j\in\{1,2,3,4\}} \frac{1}{\text{waiting time in junction}j}$ so that proceeding the traffic flow in the junction with the heaviest traffic is encouraged. We used the inferface of 2×2 junctions provided by SUMO-RL (Alegre, 2019), where every junction has 4 horizontal lanes and 4 vertical lanes. New vehicles are randomly generated with a probability of 0.1 at the end of each lane for every second.



(a) Visualization of an example state of 4 junctions. (b) 4 available actions in each junction.

Figure 8. Visualization of SUMO environment.

Baseline methods CMC, MVTCAE, SLAC, and Vanilla-RL (SAC).

Experiment setup Similar to Section D.1, following 3 steps are applied to all representation learning methods:

1. Collect a trajectory dataset composed of complete-view observations and actions by training an SAC policy from scratch for 1 million timesteps.

2. Pretrain each method across 7 seeds (0∼6) for 50 epochs using the precollected dataset.

3. For every fixed number of missing views (0∼2), train SAC for 1 million timesteps on top of the frozen latent state extracted from the pretrained method with the corresponding seed. The accumulated rewards of each method is averaged over 7 seeds (0∼6).

In SUMO, we did not split the data since it has a relatively small number of samples compared to the dataset in Bipedal Walker. We saved the weights of representation methods after training for 50 epochs and fixed them for training SAC based on these representations. Vanilla-RL(m) directly trained SAC with multi-view observations from the environment filling the missing view observations with mean values of the precollected dataset. For the given fixed number of missing views, the views to be missing were uniformly and randomly chosen in every timestep. Detailed information on the hyperparameter settings can be found in Table 2.

Hyperparameter	Value
Number of Views (N)	4
Policy	SAC
batch size	100
Replay buffer size	1,000,000
Gamma	0.99
Entropy coefficient	0.01
Number of hidden layers (all networks)	2
Number of hidden units per layer	100
Target smoothing coefficient	0.005
Target update interval	1
Temperature of relaxed categorical	0.1
Learning rate (actor & critic & representation)	3e-4
Subsequence length (H)	10
Size of the latent state (representation)	84

Table 2. The summary of hyperparameters in SUMO

D.3. Metaworld

In order to see if our method accelerates policy optimization when jointly trained with the policy, we conducted 8 complex robotic arm manipulation tasks in Metaworld (Yu et al., 2020) following the evaluation environment and protocols from Chen et al. (2021). Each task has 50 randomized configurations such as initial poses of robot arms, objects, and goals. 3 third-person-view cameras in different poses are used in each task to observe the robot arm and the objects. Since the state of the gripper attached at the end of the robot arm might not be visible in any of these three cameras, we used the ground-truth binary indicator which indicates whether the gripper is open or closed. This indicator is concatenated to the learned latent state and fed into the policy network.

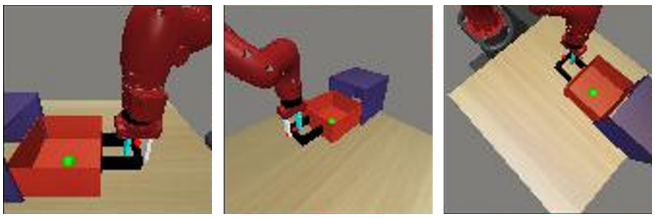


Figure 9. Visualization of 3 third-person-view observations in the task Close drawer of Metaworld environment.

Baseline methods In the complete-view scenario, we additionally employed a set of various pixel-based RL algorithms such as (1) CURL (Laskin et al., 2020b), (2) RAD (Laskin et al., 2020a), (3) Keypoint3D (Chen et al., 2021), and (4) LookCloser (Jangir et al., 2022) besides CMC, MVTCAE, SLAC, and Vanilla-RL (PPO). Brief explanations on these methods can be found in Section 2. In the missing-view scenario, however, we only included LookCloser, since we were not able to find straightforward extensions to the other baselines. We also excluded the Vanilla-RL method from our baseline methods, since we do not have a precollected dataset to compute mean values for replacing the missing-view observations.

Following Lee et al. (2020a), we applied image augmentation techniques to CMC, Keypoint3D, CURL, and RAD, but not for VAE-based methods such as MVTCAE, SLAC, and F2C. As a result, these VAE-based methods have different shapes for the input images. To handle image views in different shapes, we applied the same CNN architectures to MVTCAE, SLAC, and F2C different from CMC, CURL, RAD, and Keypoint3D while preserving the same dimensions of output features to be 128. Otherwise, we removed all the bells and whistles equally applicable to all methods such as training reward models in SLAC (Lee et al., 2020a) or utilizing empty scene of each task in Keypoint3D (Chen et al., 2021).

Experiment setup Following Chen et al. (2021), we jointly trained each representation method and PPO whose hyperparameters are determined by Chen et al. (2021). As a result, our objective function in Metaworld is as below:

$$\sum_{t=\tau\text{-}H+1}^{\tau} TC_{\text{MV-SSM}}(t;\theta) + \mathbb{E}[Q(\hat{s}_\tau, a_\tau) \text{ - } \log \pi(a_\tau|\hat{s}_\tau)], \tag{25}$$

where Q is the state-action value function that learns to minimize the soft Bellman residual and the expectation in the last term is with respect to $p(\vec{o}_0)p_\theta(\hat{s}_0|\vec{o}_0)\prod_{t=1}^{\tau} p_{\pi_{\text{old}}}(a_{t\text{-}1}|\hat{s}_{t\text{-}1})p_{\pi_{\text{old}}}(\vec{o}_t|\vec{o}_{t\text{-}1}, a_{t\text{-}1})p_\theta(\hat{s}_t|\vec{o}_t, \hat{s}_{t\text{-}1}, a_{t\text{-}1})$.

We extended SLAC to the MVRL setting by adopting the model structure from MV-SSM in F2C and optimizing Eq. (25) by replacing the first term with Eq. (7). Following Chen et al. (2021), we also employed a buffer to keep sequences of observations and actions to train MV-SSM only for a few steps ahead of each policy gradient update. Algorithm 1 summarizes the overall optimization process. Lastly, all the experiment results in Metaworld are averaged over 7 seeds (0∼6).

Algorithm 1 Fuse2Control

Input: N_{Repeat} # of iterations to repeat entire processes.
$N_{\text{MV-SSM}}$ # of steps to train MV-SSM.
B batch size, T rollout length, H horizon length.
Initialize D_{Obs}
for iter $= 1$ **to** N_{Repeat} **do**
  Initialize D_{Rollout}.
  for $b = 1$ **to** B **do**
    Run policy $\pi_{\theta_{\text{old}}}$ to collect $(\vec{o}, a, r)_{1:T}$
    $D_{\text{Rollout}} \leftarrow D_{\text{Rollout}} \cup (\vec{o}, a, r)_{1:T}$
    $D_{\text{Obs}} \leftarrow D_{\text{Obs}} \cup (\vec{o}, a)_{1:T}$
  end for
  for $i = 1$ **to** $N_{\text{MV-SSM}}$ **do**
    Sample subsequence $(\vec{o}, a)_{\tau-H+1:\tau} \sim D_{\text{Obs}}$
    Train MV-SSM by optimizing Eq. (10)
  end for
  Estimate advantage values $\hat{A}_{1:T,1:N}$ on D_{Rollout}
  for $t = 1$ **to** T **do**
    Sample subsequence $(\vec{o}, a, r)_{\tau-H+1:\tau} \sim D_{\text{Rollout}}$
    Jointly train π and MV-SSM by optimizing Eq. (18)
  end for
  $\pi_{\text{old}} \leftarrow \pi$
end for

E. Additional Experiment Results

E.1. Bipedal Walker

E.1.1. Results from another axis

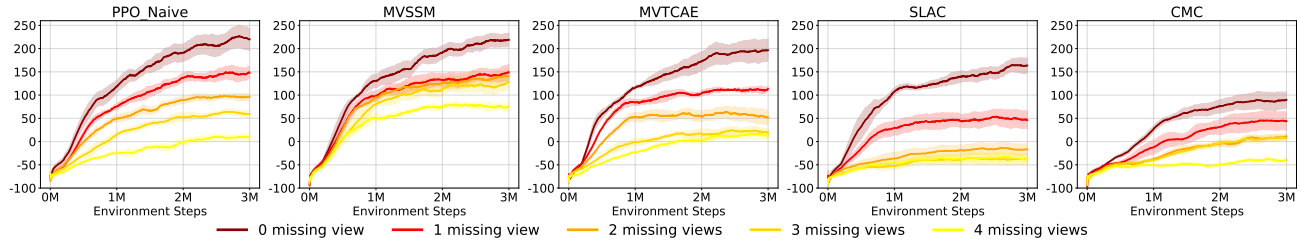


Figure 10. The performance of each method in Bipedal Walker.

To give a better sense of how robust each method is, we plot the performance of each model with a varying number of missing views in Figure 10.

E.1.2. Qualitative results

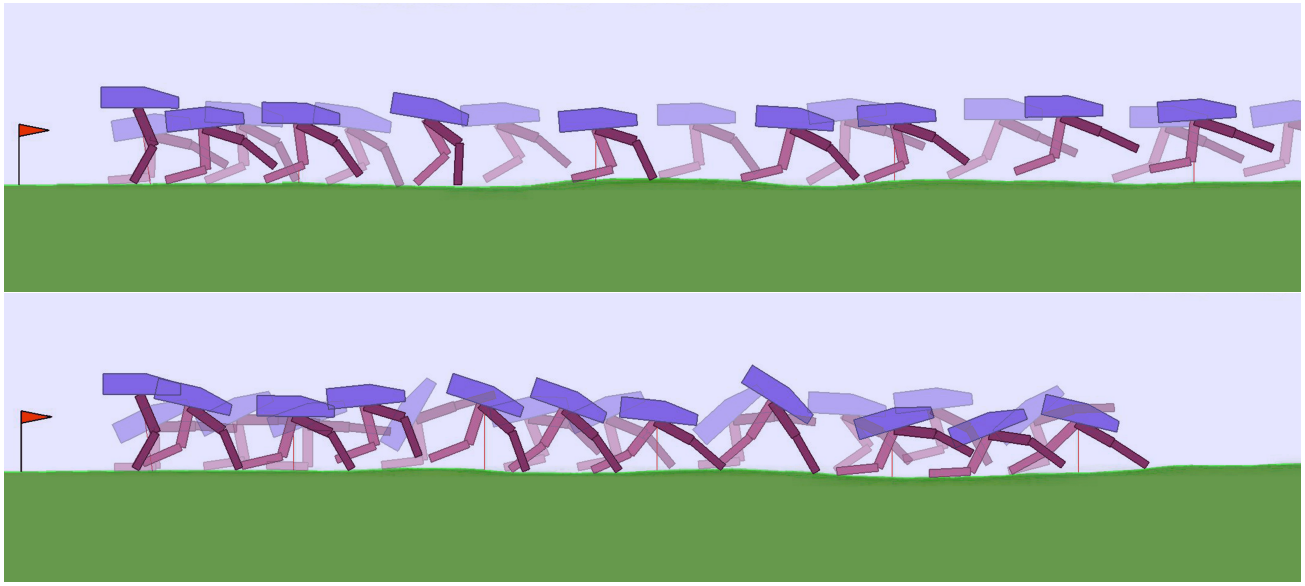


Figure 11. Visualization of trajectories from two different agents. An agent trained in the complete-view scenario (top) and An agent trained in the 1-missing-view scenario (bottom).

As all the methods including ours show noticeable decrease in their performance when 1 view is missing, we investigate in Figure 11 the behaviours of two policies trained on top of the frozen latent state from MV-SSM in Section 5.1; one is the policy trained with complete views (top row) and the other is the policy trained with 1 missing view randomly chosen per timestep (bottom row). In Figure 11, we visualize the agents every 25 steps out of a total of 500 steps. The accumulated rewards of the visualized episodes of the agents with complete views (top row) and 1 missing view (bottom row) are approximately 280 and 153, respectively.

Figure 11 shows that the policy trained with complete views barely hits the ground with its knees, which allows the agent to run faster. In contrast, the other policy trained with 1 missing view frequently hits the ground with its knees, which results in slower motion. However, such suboptimal behaviour allows the agent to avoid falling down, which prevents large negative reward. As a result, the agent prefers the safe behaviour when the fine-grained control of the agent is not feasible due to the information loss induced by missing views.

E.1.3. Different choice of the dimensions of the latent state

To see the impact of the size of the representation dimensions, we additionally conducted an experiment with the 12-dimensional representation, which is half of our original choice of the size.

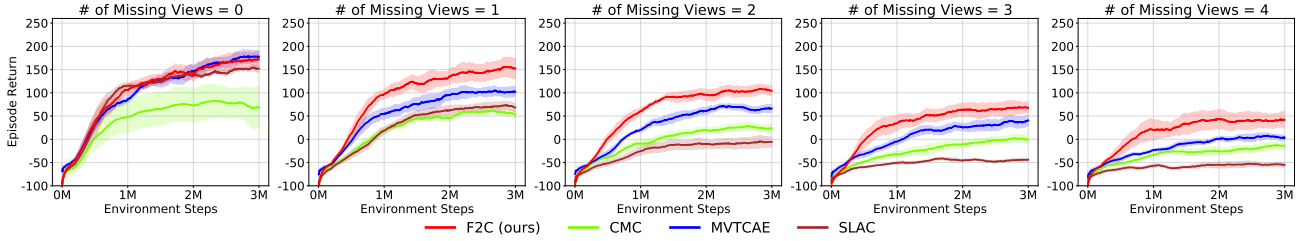


Figure 12. The performance in Bipedal Walker with 12 dimensional representation.

Figure 12 summarizes the result. Although our method still shows the most robust performance to the missing views, it shows a noticeable performance drop compared to the result of using 24-dimensional representation. We hypothesize that this is because of the contrastive loss since it enforces all the positive pairs of $(\langle \hat{s}_t, a_t \rangle, \hat{s}_{t+1})$ to be distinguishable from the negative pairs. Considering that some of the positive and negative pairs in the minibatch can be from the same trajectory, distinguishing them only with 12-dimensional information might be limited.

E.1.4. Dropout over the views

To see if regularizing over-reliance on some views with our objective function (Eq.10) has clear advantages over simple dropout regularization, we applied dropout to pretraining CMC and SLAC. We found that applying dropout to F2C and MVTCAE had negligible or negative effects, as they already have mechanisms to avoid over-reliance on some views in their objective function. We also observed that applying dropout (along with average pooling) did not improve Vanilla-RL, as average pooling discards view-specific information when employed without any auxiliary objective other than the expected return. To improve the performance of CMC and SLAC with dropout, we searched for the optimal drop rate among [0.2, 0.4, 0.6] for each method.

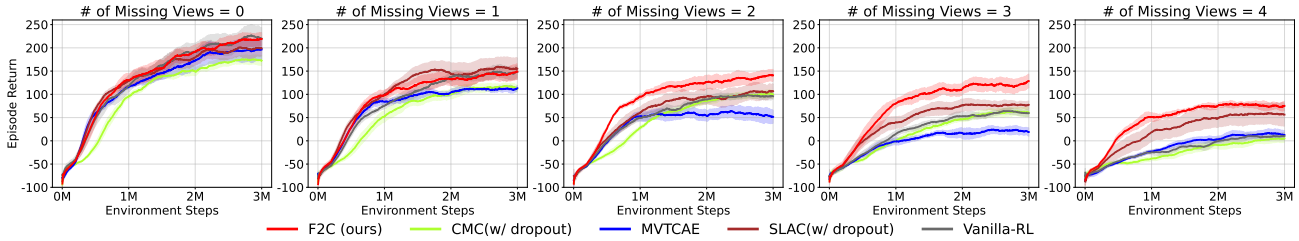


Figure 13. The performance in Bipedal Walker under missing views. From the left to the right, we increase the number of missing views. We replace two baseline methods CMC and SLAC with CMC+Dropout and SLAC+Dropout respectively.

Figure 13 shows the results in the Bipedal Walker environment. We observed that dropout effectively improved both SLAC and CMC. While SLAC and F2C perform similarly when all views are available or only one view is missing, F2C clearly outperforms SLAC when multiple views are missing. Although SLAC shows strong performance when only one view is missing, its performance noticeably drops when at least one view is additionally missing, whereas F2C remains robust to missing views. The performance of CMC is still not comparable to F2C, regardless of the number of missing views.

E.2. SUMO

E.2.1. Dropout over the views

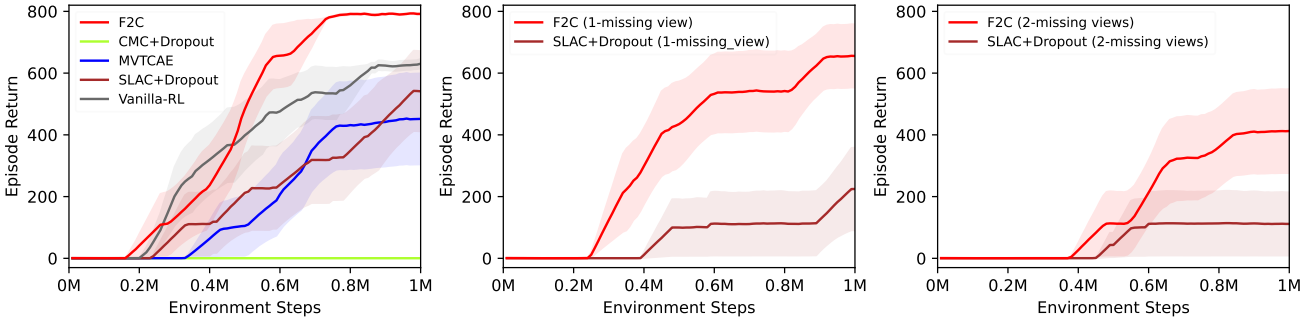


Figure 14. The performance in SUMO with complete views (left), 1-missing view (middle), and 2-missing views (right). We replace two baseline methods CMC and SLAC with CMC+Dropout and SLAC+Dropout respectively.

As in Section E.1.4, we applied dropout to pretraining CMC & SLAC and searched for the optimal drop rate among [0.2, 0.4, 0.6]. Figure 14 presents the results in the SUMO environment. We found that CMC with dropout still fails to encode informative representations for the training policy, whereas dropout improved SLAC in missing-view cases. However, SLAC fails to preserve its performance when at least one view is missing, whereas F2C performs much more robustly to missing views, reasonably preserving its performance. Overall, our results indicate that regularizing over-reliance on some views with our objective function (Eq.10) has clear advantages over the simple dropout regularization.

E.2.2. Benign sequence with fixed missing views

In our previous experiments (Section 5, Section E.1, and Section E.2.1), we focused on scenarios where the availability of views varied randomly at each timestep, resulting in missing-view observations. However, in this section, we also explore a different type of missing-view pattern referred to as the "benign" scenario, where the availability of views remains constant across all timesteps. We formally define these two missing-view patterns, utilizing the notation of missing-view observation as below:

1. Random missing views: Each view has varying availability for every timestep. For example, an observation from v-th view available at timestep t ($o_t^v \in \tilde{o}_t$) may or may not be available in the next timestep ($o_{t+1}^v \in \tilde{o}_{t+1}$ or $o_{t+1}^v \notin \tilde{o}_{t+1}$). The same rule applies to an observation from any other view v' that was not available in timestep t ($o_t^{v'} \notin \tilde{o}_t$).
2. Fixed missing views: Each view has fixed availability throughout the trajectory. Specifically, an observation from v-th view available at timestep t ($o_t^v \in \tilde{o}_t$) must be also available in the next timestep ($o_{t+1}^v \in \tilde{o}_{t+1}$), while an observation from any other view v' that was not available in timestep t ($o_t^{v'} \notin \tilde{o}_t$) must not be available in the next timestep either ($o_{t+1}^{v'} \notin \tilde{o}_{t+1}$).

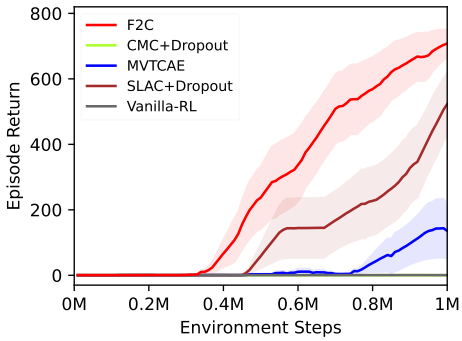


Figure 15. The performance in SUMO with dropping the last view.

We conducted an additional evaluation in SUMO with a fixed missing view, which can be considered as a more realistic scenario since using fewer sensors to reduce costs is a common practice. Specifically, based on the representations pretrained

with complete views as before, we trained the policy with dropping the last view, since all views are equally important in SUMO. Figure 15 shows the results, indicating that the constant availability of views improved the performance of SLAC and MVTCAE, while Vanilla-RL and CMC were unaffected. However, the results also demonstrate that our method, F2C, outperforms all baseline methods, indicating its robust performance not only under random missing-view patterns but also in more benign scenarios.

E.2.3. RECONSTRUCTION ERROR OF FIXED MISSING VIEWS

To further investigate the quality of learned representations of F2C, SLAC, and MVTCAE, we measured the error in predicting missing views given various missing-view combinations. To compute the error in unseen trajectories, we collected a new dataset following the same protocol in Section 5.2. For every possible subset of views, we treated it as a set of missing views that were not available across all trajectories for all timesteps. We then extracted the joint representation of available views and fed it to the decoders of the missing views to infer each missing view.

Method	1 missing view	2 missing views	3 missing views	Avg. of all cases
F2C (ours)	**0.227** $\pm$ 0.021	**0.230** $\pm$ 0.011	**0.273** $\pm$ 0.011	**0.248** $\pm$ 0.008
SLAC+Dropout	0.559 $\pm$ 0.115	0.530 $\pm$ 0.062	0.471 $\pm$ 0.045	0.508 $\pm$ 0.037
MVTCAE	4.821 $\pm$ 0.527	5.720 $\pm$ 0.410	8.474 $\pm$ 0.740	6.770 $\pm$ 0.466

Table 3. Reconstruction error with the fixed missing views.

Table 3 summarizes the results. The second, third, and fourth columns show the average error over all subsets with the same subset size, and the last column reports the average error across all cases. The table clearly demonstrates that our method has superior capability in reconstructing missing views and robustness to the increasing number of missing views. This result implies that optimizing per-view encoders with CVIBs in our objective function yields better joint representations, which is the IVW average of per-view representations. On the other hand, SLAC underperforms in all missing-view scenarios, implying that implicit optimization of per-view encoders via dropout is limited. Lastly, MVTCAE shows poor performance since inferring missing views based only on the current observations is limited when dependency across views is weak.

E.3. Evaluation in Metaworld with Complete Views

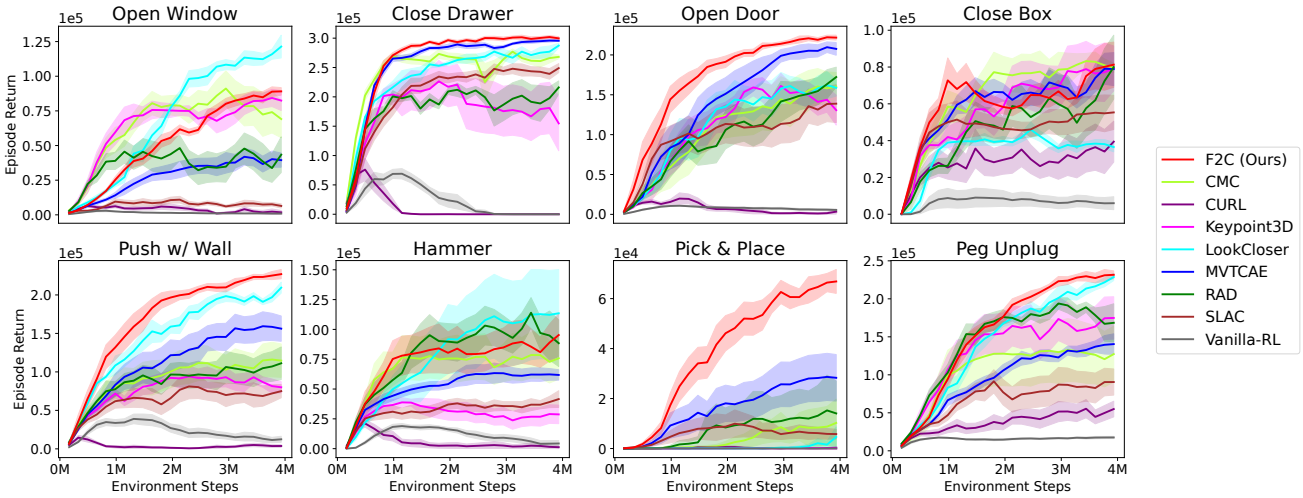


Figure 16. The performance in manipulation tasks given complete observations.

Baseline methods We introduce a set of various pixel-based RL algorithms such as (1) CURL (Laskin et al., 2020b), (2) RAD (Laskin et al., 2020a), (3) Keypoint3D (Chen et al., 2021), and (4) LookCloser (Jangir et al., 2022) in addition to CMC, MVTCAE, SLAC, and Vanilla-RL (PPO). A brief explanation on these methods can be found in Section 2.

Experiment setup Following Chen et al. (2021), we evaluated the performance of the policies jointly trained with representations of comparing methods and ours to see if our method successfully captures task-relevant information in the latent state given high dimensional image observations.

Results Figure 16 summarizes the results on learning from complete views. Among 8 control tasks, our method outperforms all the baseline methods in 5 tasks (`Close Drawer`, `Open Door`, `Push Wall`, `Pick & Place`, `Peg Unplug`) and performs on par with the best performing method in 2 tasks (`Close Box`, `Hammer`), while underperforming in 1 task (`Open Window`). On average of all tasks, the result clearly shows that our method has better sample efficiency exhibiting steeper and steadier learning curves. On the other hand, Vanilla-RL barely shows some sign of learning throughout training episodes, which implies that learning to extract meaningful information without any auxiliary supervision poses a significant bottleneck in learning the optimal control. Interestingly, LookCloser outperforms all methods in 1 task (`Open Window`) and shows competitive performance in 3 tasks (`Push Wall`, `Hammer`, `Peg Unplug`). The result implies that cross attention based on Trasnformer (Vaswani et al., 2017) encoders can be effective when complete views are given, although it shows degenerate performance give missing views as we observed in Section 5.3. Although RAD and Keypoint3D show competitive performance in `Hammer` or `Open Window`, their performances are limited in other tasks.

Ablation study MVTCAE and SLAC are special cases of our F2C framework as we discussed in Section 5.1. As shown in Figure 16, our method clearly outperforms MVTCAE in all tasks except `Close Box`, which clearly shows the advantage of learning transition dynamics in complex manipulation tasks. However, SLAC shows degenerate performance across all tasks, indicating the importance of regularizing per-view encoders.

F. Computation Resources

For Bipedal Walker, we used 40 CPU instances (n1-highcpu-32) from Google Cloud Platform (GCP).

For SUMO and Metaworld, we used 10 systems equipped with following devices.
CPU: Intel(R) Core(TM) i7-9700K CPU @ 3.60GHz
Memory: 32 GB
GPU: Nvidia TITAN V (Driver version: 440.44 & CUDA version: 10.2)

G. Note

G.1. Societal Impact

Given smaller number of sensors, we expect that our method can be positively used to reduce environmental waste and carbon footprint by matching its performance to expensive sensor systems (e.g. autonomous driving vehicles, factory automation sensors, etc.). However, we also see the possibility for our method to raise security issues. Specifically, treating each source of information allowed to be accessed as an available view and each source of the information prohibited to be accessed as a missing view, one might be able to uncover the private information by inferring the missing views using our method.

G.2. Limits

Although our method effectively handles incomplete multi-view data, it still requires all available views in every timestep to be aligned. Thus, we leave as future work the multi-view learning from a set of independent single-view datasets.

G.3. License

We used the Metaworld (Yu et al., 2020) environment and the official implementation of Keypoint3D (Chen et al., 2021), which are licensed under the MIT License.

G.4. New Assets

In the environment Bipedal Walker and SUMO, we collected datasets on our own to pretrain representations (latent states) of comparing methods and ours. We provide their anonymized links below.

Bipedal Walker: https://zenodo.org/record/6583263, https://zenodo.org/record/6583291

SUMO: https://zenodo.org/record/7568625#.Y9EtznZByHs