Representation Learning for Model-Based Reinforcement Learning: A Survey

NEW JUN JIE, National University of Singapore, Singapore

Advancements in reinforcement learning and deep learning have surpassed human experts and achieved impressive feats on many complex sequential decision making problems. To tackle the high sample complexity of reinforcement learning methods, model-based reinforcement learning learns a model of the environment dynamics to improve the sample efficiency of learning. The learnt model also enables future prediction for forward planning to improve task performance. However, from the complex and high-dimensional nature of problems emerges a trade-off between the model's sample efficiency and prediction accuracy. Representation learning methods are proposed to learn compressed yet informative latent representations of the state space. This survey aims to provide an overview of representation learning methods for learning and planning in model-based reinforcement learning, and discuss challenges and future directions from the present onward.

Additional Key Words and Phrases: Latent Variable Models, Directional Dynamics Models, Contrastive Learning

1 INTRODUCTION

"Reinforcement learning is learning what to do–how to map situations to actions–so as to maximize a numerical reward signal" [40]. Aided by the highly expressive deep neural networks, reinforcement learning has shown to be effective across a range of highly complex sequential decision making problems, such as long-horizon control and dexterous manipulation in robotics.

However, reinforcement learning faces a problem of high sample complexity, where a large number of training samples are required, limiting its application in important domains. Model-based reinforcement learning is a promising approach, by learning an explicit model of environment dynamics, in the form of a state-space model, to reduce the need for real-world samples. While model-based methods have been successful in improving sample complexity [18, 30], many unresolved challenges remain, such as the high dimensionality of observations limiting training efficiency and performance. Representation learning methods are thus proposed to learn a compressed yet informative latent representation of the state space.

Various surveys have addressed model-based methods for learning and planning [31, 33, 36, 47]. Plaat et al. (2020) surveyed the use model-based planning for high-dimensional problems [34]. Plaat et al. (2021) surveyed the challenge of high-accuracy dynamics models [35]. In contrast, this work is the first to focus and survey the landscape of representation learning methods for model-based reinforcement learning.

Section 2 provides the necessary background and formalism for reinforcement learning, model-based reinforcement learning and state-space models. Section 3 surveys the field. Section 4 introduces benchmarks commonly used for various problems and tasks. Section 5 provides a discussion on existing challenges and proposes an outlook of the field. Section 6 concludes the survey.

2 PRELIMINARIES

2.1 Reinforcement Learning

The reinforcement learning framework consists of an agent learning from its interactions with the environment[40]. With every action a_t taken by the agent, the environment returns a state s_{t+1} and a reward r_{t+1}. Reinforcement learning problems can be formally modelled as a Markov Decision Process (MDP), a 4-tuple (S, A, T_a, R_a), where S is a set of states, A is a set of actions where $A_s \subseteq A$ is the set of actions available from state s. T_a is the transition function, where $T_a(s, s')$ is

Author's address: New Jun Jie, jnew@u.nus.edu, National University of Singapore, Singapore.

the probability that action a in state s at time t will lead to state s' at time $t+1$. $R_a(s,s')$ is the immediate reward received after transitioning from state s to state s' having taken action a.

The goal of reinforcement learning is to find the optimal policy $a = \pi^*(s)$, that gives the best action a in all states $s \in S$ that will maximise the reward. The expected sum of future rewards $V^\pi(s) = E[\sum_{t=0}^{\infty} \gamma^t R_{\pi(s_t)}(s_t, s_{t+1})]$, discounted with parameter γ over t time steps, with $s = s_0$. $V^\pi(s)$ is the value function of a state.

2.2 Model-Based Reinforcement Learning

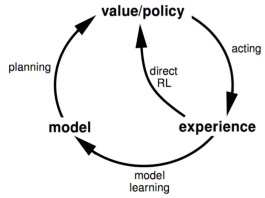

Fig. 1. Model-Based Reinforcement Learning [40]

Reinforcement learning suffers from the problem of sample inefficiency, and worsened by limited real-world data in many domains such as robotics, motivating the need for model-based reinforcement learning, having a model that models the environment dynamics, allowing sampling from the environment model, improving sample efficiency in terms of real-world data.

In model-based reinforcement learning (MBRL), a transition model $T_a(s,s')$ and optionally a reward model $R_a(s,s')$ are learnt. The models can be learned by sampling the environment and then be used to update the policy and value. When learning the transition/reward model is less complex than learning the policy model, the model-based approach is more sample efficient [34].

2.3 Latent State-Space Models

State-space models (SSMs) learn a representation of the state and can model complex non-linear transition dynamics for model-based reinforcement learning. Given a probabilistic graphical model, the joint distribution of a SSM can be factorized as:

$$p_\theta(x_{1:T}, r_{1:T}, z_{1:T}|a_{1:T}) = \Pi_{t=1}^{T} p_\theta(x_t|z_t) p_\theta(r_t|z_t) p_\theta(z_t|z_{t-1}, a_{t-1}))$$

where θ are learnable model parameters, $x_{1:T}$ denotes all observations from $t = 1, ..., T$, and likewise for $r_{1:T}$, $z_{1:T}$ and $a_{1:T}$, and the 3 distributions in the factorization correspond to observations $p_\theta(x_t|z_t)$, rewards $p_\theta(r_t|z_t)$, and transitions $p_\theta(z_t|z_{t-1}, a_{t-1})$.

The state space model can be viewed as a partially-observable Markov decision process (POMDP), where θ is learned from observed data $D = x_t, a_t, r_t{}_{t=1}^{T}$. Since maximum likelihood estimation is intractable as latent z_t's need to be marginalized out, the evidence lower bound (ELBO) under the data distribution p_d can be optimized, i.e. $E_{p_d}[L_e] \leq E_{p_d}[logp_\theta(x_{1:T}, r_{1:T}|a_{1:T})]$, where

$$L_e = \sum_{t=1}^{T} (E_{q_\phi(z_t)}[logp_\theta(x_t|z_t)] + E_{q_\phi(z_t)}[logp_\theta(r_t|z_t)] - E_{q_\phi(z_{t-1})}[D_{KL}[q_\phi(z_t)\|p_\theta(z_t|z_{t-1}, a_{t-1})]])$$

and q_ϕ is a variational distribution parameterized by ϕ.

3 SURVEY OF METHODS

The success of representation learning methods for model-based reinforcement learning depends on the problem or task targeted. The dynamics model or state-space model in model-based reinforcement learning is typically used for planning, learning and exploration tasks. In planning, the dynamics model is applied recursively to the current state to predict future states, also known as a roll-out. Multiple roll-outs are evaluated and the actions of the best roll-out is chosen. In learning, the dynamics model can generate synthetic trajectories of the data samples collected by the agent in the real environment, reducing the dependence on real-world samples to train the agent policy. In exploration, the model's prediction accuracy or uncertainty can indicate the degree to which states are unexplored.

In this section, I describe a few recent approaches of representation learning used for model-based reinforcement learning. I introduce latent variable models, directional dynamics models, contrastive representation learning, graph neural networks and causal models, and how each family of methods overcome challenges in various tasks such as planning, learning and exploration.

3.1 Latent Variable Models

Observations perceived by an agent may be high-dimensional such as images, whose distributions are difficult to model. Recent advances in deep learning have enabled learning of effective latent dynamics models [9, 15, 22, 26] that can compress high-dimensional observations into a low-dimensional yet informative abstract representation, or latent vector, of the state of the environment.

World Models was introduced by Ha and Schmidhuber (2018) [17] that compresses images using a variational autoencoder [24] to capture spatial information about the state, which is then applied to a transition function using a mixture density recurrent neural network [8] to capture temporal information of states across time. The resultant spatio-temporal latent representation is then processed by the agent instead of raw images for learning. World Models further showed that an agent that learns purely within the world model, using only model samples without any real samples, can outperform high-performing model-free reinforcement learning algorithms.

While relatively successful, learning dynamics models that are sufficiently accurate for planning in the latent space, especially in image-based domains, remain a challenge. Planning accuracy is limited by the compounding error phenomenon, where one-step prediction errors of learnt transition models can accumulate over multiple steps [3, 23, 37, 41]. To tackle accurate planning, Hafner et al. (2019) proposed the Deep Planning Network (PlaNet) [19] which showed that a dynamics model with deterministic and stochastic components can be trained with latent overshooting to minimises multi-step prediction errors, solving control tasks that exceed the difficulty of tasks previously solved by planning with learned models [4, 45].

Following the success of latent state representation and latent planning, Hafner et al. (2020) proposed Dreamer that trains both the dynamics model and the agent policy end-to-end, significantly outperforming all previous model-free and model-based methods to learn long-horizon behaviours purely within its world model, achieving human-level performance on 55 Atari games [7]. Nonetheless, while Dreamer successfully performs accurate long-term predictions up to 45 steps forward in latent space, recent model architecture advancements with transformers show improvements up to 100 steps in model prediction [21].

3.2 Directional Dynamics Models

An adjacent approach to resolving the compounding error phenomenon are dynamics models that use forward, backward and inverse models. The forward model $s_{t+1} = f(s_t, a_t)$ predicts the next state s_{t+1} given a current state s_t and action a_t at time t. The backward model $s_t = f(s_{t+1}, a_t)$

predicts the previous state given the current state and previous action taken. The inverse model $a_t = f(s_t, s_{t+1})$ predicts the action that takes the current state to the next state. To improve planning accuracy, combinations of directional dynamics models were proposed to better capture the context of a state within its latent representation.

Lai et al. (2020) showed with Bidirectional Model-based Policy Optimisation (BMPO) that constructing an additional backward dynamics model alongside a forward model lowers reliance on the accuracy of predictions of the forward model [27]. Since BMPO uses both forward and backward models to generate shorter roll-outs from real-world samples, prediction errors have fewer steps to compound, and BMPO outperforms existing model-based methods in sample efficiency and performance. In contrast to implicitly learning the context of a state, Lee et al. (2020) proposed the Context-aware Dynamics Model (CaDM) that learns an explicit context latent vector that captures contextual information useful for both forward and backward prediction.

3.3 Contrastive Representation Learning

Latent variable and directional dynamics models mentioned in previous sections require prediction of possibly high-dimensional observations, needing to reconstruct all pixels of an image-based observation. Pixel-based reconstruction is limited, especially when a small pixel change in the environment is crucial to performing well in a task, such as a small bullet on the screen. In the real world, observations also tend to be visually complex, and pixel reconstruction will not generalise well to similar tasks but with different appearances. Contrastive representation learning is an info-theoretic approach to model a latent representation based on the mutual information between variables, thus capturing only informative features of the observation, circumventing problems with pixel-based reconstruction.

Contrastive learning is a self-supervised learning method that learns useful representations without the need for labels [13, 14, 16, 20, 48], which has shown to close the performance gap between supervised and unsupervised methods of deep image models [10, 11]. Contrastive learning contrasts between samples by pulling together similar "positive" samples and pushing away different "negative" samples in the embedding space, and the resulting representation vector has shown to be highly performant for downstream tasks.

Srinivas et al. (2020) proposed Contrastive Unsupervised Representations for Reinforcement Learning (CURL) that extracts high-level features from raw pixels using contrastive learning to perform off-policy control on top of the extracted features [28]. CURL first performs data augmentations, such as random cropping, to obtain data-augmented "views" of the same image that are labelled as positive samples and other images are labelled as negative samples for contrastive learning. CURL showed that contrastive learning of augmented views nearly matches the sample efficiency of methods that use state-based features, and significantly outperformed PlaNet, Dreamer and other model-free methods on the DeepMind control (DMControl) suite.

Other than contrasting between augmented views of the same image to represent states, following the effectiveness of contrastive representations, a range of contrastive formulations has emerged. Ma et al. (2020) contrasts between an image and its learnt latent representation [30], Nguyen et al. (2021) contrasts between images of adjacent timesteps [32], Stooke et al. (2021) contrasts between images in the same trajectory [39] and Yarats et al. (2021) contrasts between data-augmented views of images at adjacent timesteps, assigning representations to prototypes before contrasting the assigned prototypes to accelerate exploration [46]. Nonetheless, Rakelly et al. (2021) proves that a representation that maximises only forward information that maximises predictive power over future states is sufficient for optimal control under any reward function [38].

3.4 Graph Neural Networks and Causal Models

Beyond problems of compounding prediction errors and high-dimensionality of observations, tasks that require compositional reasoning and counterfactual reasoning [2] has resulted in the proposal of alternative approaches to tackle them. Compositional reasoning allows agents to reason about a state in terms of its individual, localized elements. Counterfactual reasoning allows agents to reason about possible alternatives beyond what is observed through interventional actions. Graph neural networks tackle compositional reasoning by disentangling scenes into objects, their properties and relations between them [5, 12]. Kipf et al. (2019) proposed Contrastive learning of Structured World Models (C-SWM), applying a graph neural network model to predict updates in the latent state representation between the current and the next states $\Delta_{z_t} = T(z_t, a_t) = GNN(\{(z_t^k, a_t^k)\}_{k=1}^K)$ [25]. C-SWM shows that an object-based factorization of the state provides a strong inductive bias that allows for an agent to generalise to novel interactions of objects. To perform counterfactual reasoning, Li et al. (2020) proposes Causal World Models (CWM) that applies graph neural networks similar to C-SWM, but in contrast, additionally models the relationship between intervened observations and alternative futures by estimating latent confounding factors. CWM demonstrates superior performance to C-SWM on physical reasoning tasks [29].

4 BENCHMARKS

To evaluate the diversity of tasks and problems involving model-based reinforcement learning, a wide variety of benchmarks for model-based reinforcement learning exists, with commonly used ones being MuJoCo, a physics machine for model-based control in robotics [43], the Arcade Learning Environment (ALE), an interface to hundreds of Atari 2600 game environments designed to be a challenge for human players [7], the DeepMind Control Suite (DMControl), a set of continuous control tasks intended as benchmarks for reinforcement learning agents [42], the DeepMind Lab (DMLab), a 3D game platform for complex tasks in large, partially-observed and visually diverse worlds [6], and Robosuite, a simulation framework for robot learning powered by MuJoCo [50].

The choice of benchmarks for model-based reinforcement learning methods depend on the task that the method aims to tackle or the problem that the method aims to solve. DMControl was chosen to evaluate Dreamer for learning long-horizon behaviours and credit assignment. C-SWM was evaluated on simple block pushing environment to understand the accuracy of its learnt latent transitions. Given the highly diverse nature of sequential decision making tasks, benchmarks chosen need to be specific to the task or problem targeted. Tasks may vary significantly, such as control tasks with a different degrees of freedom, long-horizon tasks that require long-term planning, tasks with visually complex observations and tasks that require combinatorial or causal reasoning. For example, the MuJoCo environment's backgrounds were replaced with natural videos to evaluate CVRL on performance on complex visual observations. Problems are similarly diverse, with continuous action spaces, partial observability of the environment, multiple interacting cooperative or competitive agents, imperfect or asymmetric information between agents and stochastic environments, and will require custom modifications to existing benchmarking environments.

5 CHALLENGES AND OUTLOOK

5.1 Reproducibility

Deep reinforcement learning agents suffer from multiple critical challenges, such as high variance in training, hyperparameter sensitivity of results and heavy computational requirements, which may significantly hinder the reproducibility of methods proposed. Deep reinforcement learning agents suffer from high variance in training because of chance exploration, where agents may stumble upon a high-scoring state sometimes and miss out on such states at other times, resulting in a high

variance in final performance across different random seeds. The performance of deep reinforcement learning methods are generally hyperparameter-sensitive, and one particular configuration of hyperparameters that is optimised for one benchmark may not be sufficiently optimised for other benchmarks. Since model-based reinforcement learning includes several components–training of a dynamics model, the prediction of trajectories, policy optimization and planning–each component requires the tuning of several design parameters that can significantly impact performance [49]. Due to the heavy computationl requirements of training deep reinforcement learning agents, and a large permutation of hyperparameter configurations to tune, worsened by the need for a sufficient number of random seeds to adjust for variance, without the exact code development setup, reproducibility of published methods' performance for comparison with newer algorithms is highly challenging, which is exacerbated by the lack of open-source code [44]. Agarwal et al. (2021) show discrepancies in the comparisons of methods in previous works due to the statistical uncertainty implied by the use of a limited number of training runs [1]. Nonetheless, a needed change in how performance is evaluated would prevent unreliable results from stagnating the field.

5.2 Benchmarking

Aside from reproducibility, benchmarking is a complex challenge in model-based reinforcement learning because tasks and problems are highly diverse. The improvement sought by model-based reinforcement learning are not limited to sample efficiency and planning accuracy but also the interpretability of agent policies, exploration efficiency and many others. Since benchmarking environments need to be targeted to specific tasks and problems, non-standard modifications are required of environments, such as pre-processing of observations, reward function modifications or different episode horizons [44], further worsening the difficulty to make accurate comparisons across methods and quantify scientific progress. Nonetheless, benchmarking challenges seem to be reflective of the high complexity nature of sequential decision making problems in general.

5.3 Outlook

Model-based reinforcement learning has shown to be promising as it can exceed their model-free counterparts in performance and sample efficiency [34], and the model-based framework lends itself well to counterfactual reasoning. Given the improvements in representation learning, which has shown to further improve model-based reinforcement learning methods, further advancements in representation learning are likely to further drive performance improvements. Representation learning for model-based reinforcement learning will increasingly be used for tasks not just in planning and learning but for exploration, interpretability, counterfactual reasoning and compositional learning, among many others. Despite the complex challenges of model-based reinforcement learning research, a focus on the robustness of experimentation methodology will lead to an optimistic outlook for the field of model-based reinforcement learning.

6 CONCLUSION

Model-based reinforcement learning has benefited greatly from advancements in representation learning. The new approaches allow us to approach more complex and diverse problems and tasks than before. The literature of representation learning for model-based reinforcement learning is widely varied, with latent variable models, directional dynamics models, contrastive representation learning, graph neural networks and causal models. This survey is a brief summary of the literature on representation learning for model-based reinforcement learning, and there exists other alternative approaches that are likely to have also been applied on model-based reinforcement learning. I hope that this survey will contribute to a better overview of the diverse approaches and considerations to representation learning for model-based reinforcement learning.

REFERENCES

[1] Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron Courville, and Marc G Bellemare. 2021. Deep reinforcement learning at the edge of the statistical precipice. *arXiv preprint arXiv:2108.13264* (2021).

[2] Ossama Ahmed, Frederik Träuble, Anirudh Goyal, Alexander Neitz, Yoshua Bengio, Bernhard Schölkopf, Manuel Wüthrich, and Stefan Bauer. 2020. Causalworld: A robotic manipulation benchmark for causal structure and transfer learning. *arXiv preprint arXiv:2010.04296* (2020).

[3] Kavosh Asadi, Dipendra Misra, and Michael Littman. 2018. Lipschitz continuity in model-based reinforcement learning. In *International Conference on Machine Learning*. PMLR, 264–273.

[4] Ershad Banijamali, Rui Shu, Hung Bui, Ali Ghodsi, et al. 2018. Robust locally-linear controllable embedding. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 1751–1759.

[5] Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. 2018. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261* (2018).

[6] Charles Beattie, Joel Z Leibo, Denis Teplyashin, Tom Ward, Marcus Wainwright, Heinrich Küttler, Andrew Lefrancq, Simon Green, Víctor Valdés, Amir Sadik, et al. 2016. Deepmind lab. *arXiv preprint arXiv:1612.03801* (2016).

[7] Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. 2013. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research* 47 (2013), 253–279.

[8] Christopher M Bishop. 1994. Mixture density networks. (1994).

[9] Lars Buesing, Theophane Weber, Sébastien Racaniere, SM Eslami, Danilo Rezende, David P Reichert, Fabio Viola, Frederic Besse, Karol Gregor, Demis Hassabis, et al. 2018. Learning and querying fast generative models for reinforcement learning. *arXiv preprint arXiv:1802.03006* (2018).

[10] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. 2018. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 132–149.

[11] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. 2020. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882* (2020).

[12] Michael B Chang, Tomer Ullman, Antonio Torralba, and Joshua B Tenenbaum. 2016. A compositional object-based approach to learning physical dynamics. *arXiv preprint arXiv:1612.00341* (2016).

[13] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*. PMLR, 1597–1607.

[14] Xinlei Chen and Kaiming He. 2021. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 15750–15758.

[15] Andreas Doerr, Christian Daniel, Martin Schiegg, Nguyen-Tuong Duy, Stefan Schaal, Marc Toussaint, and Trimpe Sebastian. 2018. Probabilistic recurrent state-space models. In *International Conference on Machine Learning*. PMLR, 1280–1289.

[16] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. 2020. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733* (2020).

[17] David Ha and Jürgen Schmidhuber. 2018. World models. *arXiv preprint arXiv:1803.10122* (2018).

[18] Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. 2019. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603* (2019).

[19] Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. 2019. Learning latent dynamics for planning from pixels. In *International Conference on Machine Learning*. PMLR, 2555–2565.

[20] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9729–9738.

[21] Michael Janner, Qiyang Li, and Sergey Levine. 2021. Reinforcement Learning as One Big Sequence Modeling Problem. *arXiv preprint arXiv:2106.02039* (2021).

[22] Maximilian Karl, Maximilian Soelch, Justin Bayer, and Patrick Van der Smagt. 2016. Deep variational bayes filters: Unsupervised learning of state space models from raw data. *arXiv preprint arXiv:1605.06432* (2016).

[23] Nan Rosemary Ke, Amanpreet Singh, Ahmed Touati, Anirudh Goyal, Yoshua Bengio, Devi Parikh, and Dhruv Batra. 2019. Learning dynamics model in reinforcement learning by incorporating the long term future. *arXiv preprint arXiv:1903.01599* (2019).

[24] Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

[25] Thomas Kipf, Elise van der Pol, and Max Welling. 2019. Contrastive learning of structured world models. *arXiv preprint arXiv:1911.12247* (2019).

[26] Rahul G Krishnan, Uri Shalit, and David Sontag. 2015. Deep kalman filters. *arXiv preprint arXiv:1511.05121* (2015).

[27] Hang Lai, Jian Shen, Weinan Zhang, and Yong Yu. 2020. Bidirectional Model-based Policy Optimization. In *International Conference on Machine Learning*. PMLR, 5618–5627.

[28] Michael Laskin, Aravind Srinivas, and Pieter Abbeel. 2020. Curl: Contrastive unsupervised representations for reinforcement learning. In *International Conference on Machine Learning*. PMLR, 5639–5650.

[29] Minne Li, Mengyue Yang, Furui Liu, Xu Chen, Zhitang Chen, and Jun Wang. 2020. Causal World Models by Unsupervised Deconfounding of Physical Dynamics. *arXiv preprint arXiv:2012.14228* (2020).

[30] Xiao Ma, Siwei Chen, David Hsu, and Wee Sun Lee. 2020. Contrastive Variational Model-Based Reinforcement Learning for Complex Observations. *arXiv preprint arXiv:2008.02430* (2020).

[31] Thomas M Moerland, Joost Broekens, and Catholijn M Jonker. 2020. Model-based reinforcement learning: A survey. *arXiv preprint arXiv:2006.16712* (2020).

[32] Tung Nguyen, Rui Shu, Tuan Pham, Hung Bui, and Stefano Ermon. 2021. Temporal Predictive Coding For Model-Based Planning In Latent Space. *arXiv preprint arXiv:2106.07156* (2021).

[33] Constantin-Valentin Pal and Florin Leon. 2020. Brief Survey of Model-Based Reinforcement Learning Techniques. In *2020 24th International Conference on System Theory, Control and Computing (ICSTCC)*. IEEE, 92–97.

[34] Aske Plaat, Walter Kosters, and Mike Preuss. 2020. Deep Model-Based Reinforcement Learning for High-Dimensional Problems, a Survey. *arXiv preprint arXiv:2008.05598* (2020).

[35] Aske Plaat, Walter Kosters, and Mike Preuss. 2021. High-Accuracy Model-Based Reinforcement Learning, a Survey. *arXiv preprint arXiv:2107.08241* (2021).

[36] Athanasios S Polydoros and Lazaros Nalpantidis. 2017. Survey of model-based reinforcement learning: Applications on robotics. *Journal of Intelligent & Robotic Systems* 86, 2 (2017), 153–173.

[37] Sébastien Racanière, Théophane Weber, David P Reichert, Lars Buesing, Arthur Guez, Danilo Rezende, Adria Puigdomenech Badia, Oriol Vinyals, Nicolas Heess, Yujia Li, et al. 2017. Imagination-augmented agents for deep reinforcement learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 5694–5705.

[38] Kate Rakelly, Abhishek Gupta, Carlos Florensa, and Sergey Levine. 2021. Which Mutual-Information Representation Learning Objectives are Sufficient for Control? *arXiv preprint arXiv:2106.07278* (2021).

[39] Adam Stooke, Kimin Lee, Pieter Abbeel, and Michael Laskin. 2021. Decoupling representation learning from reinforcement learning. In *International Conference on Machine Learning*. PMLR, 9870–9879.

[40] Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

[41] Erik Talvitie. 2014. Model Regularization for Stable Sample Rollouts.. In *UAI*. 780–789.

[42] Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, et al. 2018. Deepmind control suite. *arXiv preprint arXiv:1801.00690* (2018).

[43] Emanuel Todorov, Tom Erez, and Yuval Tassa. 2012. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 5026–5033.

[44] Tingwu Wang, Xuchan Bao, Ignasi Clavera, Jerrick Hoang, Yeming Wen, Eric Langlois, Shunshi Zhang, Guodong Zhang, Pieter Abbeel, and Jimmy Ba. 2019. Benchmarking model-based reinforcement learning. *arXiv preprint arXiv:1907.02057* (2019).

[45] Manuel Watter, Jost Tobias Springenberg, Joschka Boedecker, and Martin Riedmiller. 2015. Embed to control: A locally linear latent dynamics model for control from raw images. *arXiv preprint arXiv:1506.07365* (2015).

[46] Denis Yarats, Rob Fergus, Alessandro Lazaric, and Lerrel Pinto. 2021. Reinforcement learning with prototypical representations. *arXiv preprint arXiv:2102.11271* (2021).

[47] Fengji Yi, Wenlong Fu, and Huan Liang. 2018. Model-based reinforcement learning: A survey. In *Proceedings of the International Conference on Electronic Business (ICEB), Guilin, China*. 2–6.

[48] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. 2021. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv preprint arXiv:2103.03230* (2021).

[49] Baohe Zhang, Raghu Rajan, Luis Pineda, Nathan Lambert, André Biedenkapp, Kurtland Chua, Frank Hutter, and Roberto Calandra. 2021. On the importance of hyperparameter optimization for model-based reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 4015–4023.

[50] Yuke Zhu, Josiah Wong, Ajay Mandlekar, and Roberto Martín-Martín. 2020. robosuite: A modular simulation framework and benchmark for robot learning. *arXiv preprint arXiv:2009.12293* (2020).