Multimodal foundation world models for generalist embodied agents

Pietro Mazzaglia Tim Verbelen Bart Dhoedt Aaron Courville Sai Rajeswar

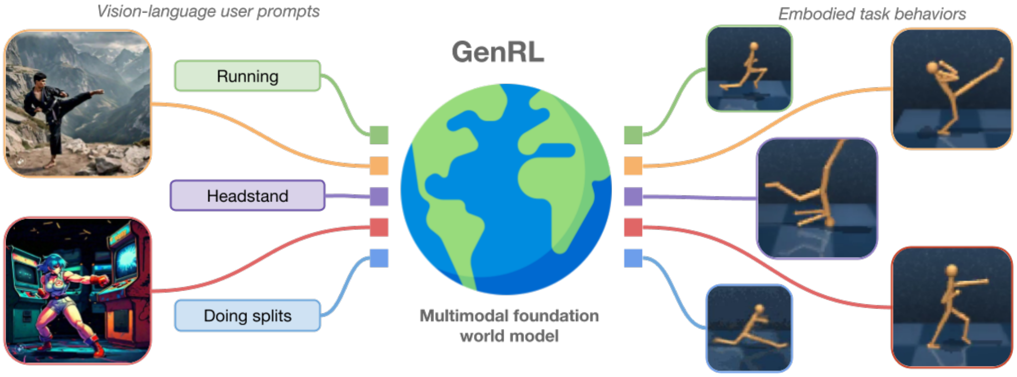

Vision-language user prompts
Running
Headstand
Doing splits
GenRL
Multimodal foundation world model
Embodied task behaviors

Figure 1: *Multimodal foundation world models* connect and align the video-language space of a foundation model with the latent space of a generative world model for reinforcement learning, requiring vision-only data. Our *GenRL* framework turns visual and/or language prompts into latent targets and learns to realize the corresponding behaviors by training in the world model's imagination.

Abstract

Learning generalist embodied agents, able to solve multitudes of tasks in different domains is a long-standing problem. Reinforcement learning (RL) is hard to scale up as it requires a complex reward design for each task. In contrast, language can specify tasks in a more natural way. Current foundation vision-language models (VLMs) generally require fine-tuning or other adaptations to be functional, due to the significant domain gap. However, the lack of multimodal data in such domains represents an obstacle toward developing foundation models for embodied applications. In this work, we overcome these problems by presenting multimodal foundation world models, able to connect and align the representation of foundation VLMs with the latent space of generative world models for RL, without any language annotations. The resulting agent learning framework, GenRL, allows one to specify tasks through vision and/or language prompts, ground them in the embodied domain's dynamics, and learns the corresponding behaviors in imagination. As assessed through large-scale multi-task benchmarking, GenRL exhibits strong multi-task generalization performance in several locomotion and manipulation domains. Furthermore, by introducing a data-free RL strategy, it lays the groundwork for foundation model-based RL for generalist embodied agents.

Project website: mazpie.github.io/genrl

1. Introduction

In most RL settings, we lack multimodal data to train or fine-tune domain-specific foundation models, due to the costs of labelling agents' interactions and/or due to the intrinsic unsuitability of some embodied contexts to be converted into language. For instance, in robotics, it's non-trivial to convert a language description of a task to the agent's actions which are hardware-level controls, such as motor currents or joint torques. These difficulties make it hard to scale current techniques to large-scale generalization settings.

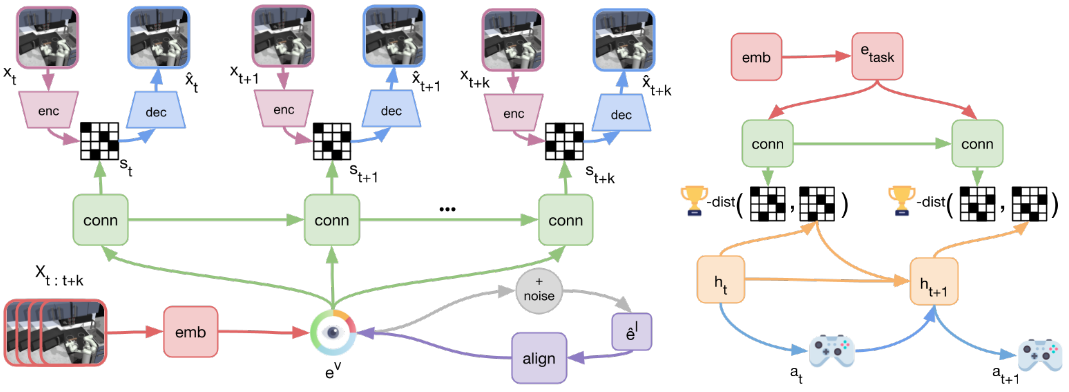
In this work, we present *GenRL*, a novel approach for training generalist agents from visual or language prompts, requiring no language annotations (Figure 1). Analogously to foundation models for vision and language, GenRL allows generalization to new tasks without additional data and lays the groundwork for foundation models in embodied RL domains (R. Bommasani et al, 2022).

2. Preliminaries

Related work can be found in Appendix A.

Problem setting. The agent receives from the environment observations $x \in \mathcal{X}$ and interacts with it through actions $a \in \mathcal{A}$. The objective of the agent is to accomplish a certain task, which can be specified either in the observation space x_{task}, e.g. through images or videos, or in language space y_{task}, where $\mathcal{Y}$ represents the space of all possible sentences. Crucially, compared to a standard RL setting, we do not assume that a reward signal is available to solve the task. When a reward function exists, it is instead used to evaluate the agent's performance.

Correspondence to: pietro.mazzaglia@ugent.be.
Affiliations and acknowledgements later, before the bibliography.
Multi-modal Foundation Model meets Embodied AI @ ICML Vienna, Austria. 2024.



(a) Connecting and aligning (Section 3.1)

(b) Learning task behavior (Section 3.2)

Figure 2: The agent learns a multimodal foundation world model that connects and aligns (a) the representation of a foundation VLM with the latent states of a generative world model. Given a certain task prompt, (b) the model allows embedding the task and translating into targets in the latent dynamics space. Then, the agent can learn to accomplish the target states by using RL in imagination.

3. GenRL

World model. GenRL learns a task-agnostic world model representation by modelling the sequential dynamics of the environment in a compact discrete latent space S (Hafner et al., 2020; 2023). Latent states $s \in S$ are sampled from independent categorical distributions. The gradients for training the model are propagated through the sampling process with straight-through estimation (Bengio et al., 2013).

The world model is made of the following components:

$$\begin{aligned}
&\text{Encoder:} && q_\phi(s_t|x_t),\\
&\text{Sequence model:} && h_t = f_\phi(s_{t-1}, a_{t-1}, h_{t-1}),\\
&\text{Dynamics predictor:} && p_\phi(s_t|h_t),\\
&\text{Decoder:} && p_\phi(x_t|s_t),
\end{aligned}$$

The sequence model is implemented as a linear GRU cell (Chung et al., 2014). The world model training loss is:

$$\begin{aligned}
\mathcal{L}_\phi = \sum_t \underbrace{D_{\mathrm{KL}}\left[q_\phi(s_t|x_t) \| p_\phi(s_t|s_{t-1}, a_{t-1})\right]}_{\text{dyn loss}} - \\
\underbrace{\mathbb{E}_{q_\phi(s_t|x_t)}\left[\log p_\phi(x_t|s_t)\right]}_{\text{recon loss}},
\end{aligned} \tag{1}$$

where $p_\phi(s_t|s_{t-1}, a_{t-1})$ is a shorthand for $p_\phi(s_t|f_\phi(s_{t-1}, a_{t-1}, h_{t-1}))$. Differently from recurrent state space models (RSSM; (Hafner et al., 2019)), for our framework, encoder and decoder models are not conditioned on the information present in the sequence model. This ensures that the latent states only contain information about a single observation. We can then leverage the encoder as a probabilistic visual tokenizer, that is grounded in the target embodied environment.

3.1. Multimodal foundation world models

Multimodal VLMs are large pre-trained models that have the following components:

$$\begin{aligned}
\text{Vision embedder:} \quad & e^{(v)} = f_{\mathrm{PT}}^{(v)}(x_{t:t+k}),\\
\text{Language embedder:} \quad & e^{(l)} = f_{\mathrm{PT}}^{(l)}(y),
\end{aligned}$$

where $x_{t:t+k}$ is a sequence of visual observations and y is a text prompt. For video-language models, k is generally a constant number of frames (e.g. $k \in \{4, 8, 16\}$ frames). Image-language models are a special case where $k = 1$ as the vision embedder takes a single frame as an input. For our implementation, we adopt the InternVideo2 video-language model (Wang et al., 2024).

To connect the representation of the multimodal foundation VLM with the world model latent space, we instantiate two modules: a *latent connector* $p_\psi(s_{t:t+k}|e)$ trained to minimize:

$$\mathcal{L}_{\text{conn}} = \sum_t D_{\mathrm{KL}}\left[p_\psi(s_t|s_{t-1}, e) \| \mathrm{sg}(q_\phi(s_t|x_t))\right],$$

where $\mathrm{sg}(\cdot)$ indicates to stop gradients propagating, and a *representation aligner* $e^{(v)} = f_\psi(e^{(l)})$, trained to minimize:

$$\mathcal{L}_{\text{align}} = \|e^{(v)} - f_\psi(e^{(l)})\|_2^2.$$

The connector learns to predict the latent states of the world model from embeddings in the VLM's representation space. The training objective consist of minimizing the KL divergence between its predictions and the world model's encoder distribution. While more expressive architectures, such as transformers (Vaswani et al., 2023) or state-space models (Gu & Dao, 2023) could be adopted, for simplicity, we stick with a GRU-based architecture, same as the world model.

Multimodal VLMs trained with contrastive learning exhibit a multimodality gap (Liang et al., 2022), where the spherical embeddings of different modalities are not aligned. The role of the aligner is to reduce this multimodality gap, by projecting text embeddings into their corresponding visual embeddings. Having a dataset of vision-language data, this projective function can be learned. In our settings, given the absence of multimodal datasets in embodied domains, we leverage the idea that language embeddings are similar to vision embeddings 'corrupted' with noise, i.e. $e^{(l)} \approx e^{(v)} + \epsilon$ (Zhang et al., 2024; Zhou et al., 2022), and so we train the aligner with noisy vision embeddings as inputs.

3.2. Learning specified task behaviors in imagination

World models can be used to train behavior policies in a model-based RL fashion (Hafner et al., 2020). Given a task specified through a visual or language prompt, our MFWM can generate the corresponding latent states by turning the embedder's output, e_{task}, into sequences of latent states $s_{t:t+k}$ (examples are shown in Figure 1). The objective of the policy model π_θ is then to match the goals specified by the user by performing trajectory matching.

Trajectory matching can be solved as a divergence minimization problem (Englert et al., 2013) between the distribution of the states visited by the policy π_θ and the ones generated using the aligner-connector networks from the user-specified prompt. For the divergence function, we found that using the cosine distance between linear projections of the latent states works well. Thus, we define the reward for RL:

$$r_{\text{GenRL}} = \cos\big(g_\phi(s^{\text{dyn}}_{t+1}), g_\phi(s^{\text{task}}_{t+1})\big), \tag{2}$$

where g_ϕ represents the first linear layer of the world model's decoder. We train an actor-critic model to maximize this reward and achieve the tasks specified by the user (Hafner et al., 2023). Additional implementation details are provided in Appendix B.

One issue with trajectory matching is that it assumes that the distribution of states visited by the agent starts from the same state as the target distribution. However, the initial state generated by the connector may differ from the initial state where the policy is currently in. To address this alignment issue, we introduced a *best matching trajectory* technique, inspired by best path decoding in speech recognition (Graves et al., 2006). Please refer to Appendix E for details about this technique and an ablation study.

4. Experiments

4.1. Offline RL

Table 1: Offline RL from language prompts on tasks that are included in the training dataset. Scores are normalized episodic rewards averaged over 5 seeds. Detailed results in Appendix F.

	Image-language reward			Video-language reward			Ours
	IQL	TD3+BC	TD3	IQL	TD3+BC	TD3	GenRL
walker (3 tasks)	0.43	0.50	0.39	0.47	0.41	0.74	**0.91**
cheetah (1 task)	0.37	-0.01	-0.01	0.15	-0.01	0.34	**0.74**
quadruped (3 tasks)	0.26	0.31	0.41	0.31	0.31	0.44	**0.77**
stickman (3 tasks)	0.38	0.48	0.26	0.42	0.37	0.41	**0.54**
kitchen (4 tasks)	0.15	0.36	0.17	0.04	0.00	0.43	**0.69**
overall	0.30	0.38	0.28	0.28	0.23	0.49	**0.73**

We aim to assess the multi-task capabilities of different approaches for designing rewards using VLMs. We collected large datasets for each of the domains evaluated, containing a mix of structured data (i.e. the replay buffer of agents learning to perform some tasks) and unstructured data (i.e. exploration data collected using (Sekar et al., 2020)). We have removed the explicit reward information about the task and replaced it with a short task description, in language form. Details about datasets, tasks, and prompts used can be found in the Appendix C

We compare GenRL to two main categories of approaches:

- *Image-language rewards*: following (Rocamonde et al., 2023), the cosine similarity between the embedding for the language prompt and the embedding for the agent's visual observation is used as a reward. For the VLM, we adopt the SigLIP-B (Zhai et al., 2023) model.
- *Video-language rewards*: similar to the image-language rewards, with the difference that the vision embedding is computed from a video of the history of the last k frames, as done in (Fan et al., 2022). The VLM is the InternVideo2 model (Wang et al., 2024).

We test both approaches with a variety of offline RL methods, including IQL (Kostrikov et al., 2021), TD3+BC (Fujimoto & Gu, 2021), and TD3 (Fujimoto et al., 2018). All methods are trained for 500k gradient steps, and evaluated on 20 episodes. Other details are reported in Appendix D.

Behavior extraction. We want to verify whether the methods can retrieve the tasks behaviors that are certainly present in the dataset. We present summarized results in Table 1, with episodic rewards rescaled so that 0 represents the performance of a random agent, while 1 represents the performance of an expert agent. GenRL excels in overall performance across all domains, particularly in the quadruped, cheetah and kitchen domains.

Multi-task generalization. To assess multi-task generalization, we defined a set of tasks not included in the training data. Although we don't anticipate agents matching the performance of expert models, higher scores in this benchmark help gauge the generalization abilities of different methods. We averaged the performance across various tasks for each domain and summarized the findings in Figure 3, with detailed task results in Appendix F.

Overall, we observe a similar trend as for the behavior extraction results. GenRL significantly outperforms the

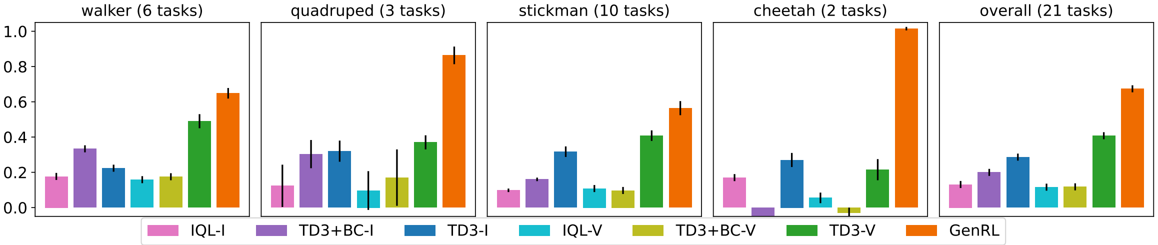


Figure 3: Offline RL from language prompts on tasks that are not deliberately included in the training dataset. Performance averaged over 5 seeds and standard error was reported with black lines. Detailed results per task in Appendix F.

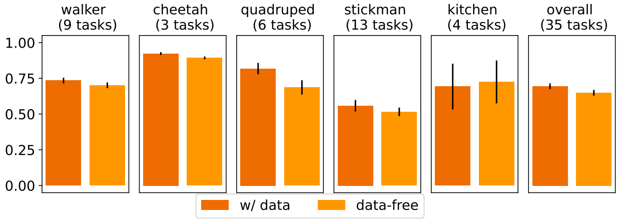


Figure 4: Learning behaviors in imagination without relying on any data for initializing the sequences to learn in imagination. Performance averaged for 5 seeds. Detailed results in Appendix F.

other approaches, with performance close to expert level in quadruped and cheetah tasks.

4.2. Data-free RL

Foundation models (R. Bommasani et al, 2022) are generally trained on enormous datasets in order to generalize to new tasks. The datasets used for the model pretraining are not necessary for the downstream applications, and sometimes these datasets are not even publicly available (OpenAI et al, 2024; Gemini Team et al, 2024). In this section, we aim to establish a new paradigm for foundation models in RL, which follows the same principle of foundation models for vision and language. We call this paradigm *data-free RL* and we define it as the ability to adapt for new tasks, after pre-training, using no additional data.

GenRL enables data-free RL thanks to two main reasons: the agent learns a task-agnostic MFWM on a large varied dataset during pre-training, and the MFWM enables the possibility of specifying tasks directly in latent space, without requiring any data. Thus, in order to learn behaviors in imagination, the agent can: (i) sample random latent states in the world model's representation, (ii) rollout sequences in imagination, following the policy, and (iii) compute rewards, using the targets obtained by processing the given prompts with the connector-aligner networks.

In Figure 4, we compare data-free RL to traditional offline RL with GenRL, as discussed in Section 4.1. While data-free RL generally shows a slight decrease in overall performance, the differences are minimal across most domains, and it even outperforms in the kitchen.

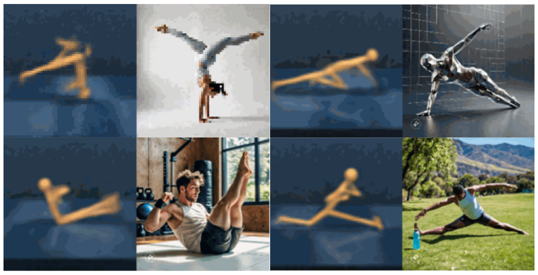


Figure 5: Video grounding examples, showing decoded latent visual targets next to their visual prompts (videos on website)

5. Additional Analysis

A framework for behavior generation. A common challenge with using LLMs and VLMs involves the need for prompt tuning to achieve specific tasks. GenRL uniquely allows for the visualization of targets obtained from specific prompts. By decoding the latent targets, using the MFWM decoder, we can visualize the interpreted prompt before training the corresponding behavior. This enables a much more explainable framework, which allows fast iteration for prompt tuning.

Video grounding. Similarly as for language prompts, GenRL allows translating vision prompts (short videos) into behaviors. In Figure 5, we present a set of video grounding examples, obtained by inferring the latent targets corresponding to the vision prompts (right image) and then using the decoder model to decode images (left image). We observe that the agent is able to translate short human action videos into the same actions but for the Stickman embodiment. By applying this approach, it would be possible to learn behaviors from a single video.

6. Discussion

We introduced GenRL, a world-model based approach for grounding vision-language prompts into embodied domains and learning the corresponding behaviors in imagination. The multimodal foundation world models of GenRL can be trained using unimodal data, overcoming the lack of multimodal data in embodied RL domains. Data-free RL with GenRL lays the groundwork for foundation models in RL that can generalize to new tasks, learning new behaviors without additional data.

Affiliations

Pietro Mazzaglia is at IDLab, Ghent University, Belgium. The work has been done while Pietro was an intern at Mila and ServiceNow Research. Tim Verbelen is at VERSES AI Research Lab, California, USA. Bart Dhoedt is at IDLab, Ghent University, Belgium. Aaron Courville is at Mila, University of Montreal, Canada. Sai Rajeswar is at ServiceNow Research.

Acknowledgments

Pietro Mazzaglia is funded by a Ph.D. grant of the Flanders Research Foundation (FWO). This research was supported by a Mitacs Accelerate Grant.

References

Baker, B., Akkaya, I., Zhokhov, P., Huizinga, J., Tang, J., Ecoffet, A., Houghton, B., Sampedro, R., and Clune, J. Video pretraining (vpt): Learning to act by watching unlabeled online videos, 2022.

Bar-Tal, O., Chefer, H., Tov, O., Herrmann, C., Paiss, R., Zada, S., Ephrat, A., Hur, J., Liu, G., Raj, A., Li, Y., Rubinstein, M., Michaeli, T., Wang, O., Sun, D., Dekel, T., and Mosseri, I. Lumiere: A space-time diffusion model for video generation, 2024.

Baumli, K., Baveja, S., Behbahani, F., Chan, H., Comanici, G., Flennerhag, S., Gazeau, M., Holsheimer, K., Horgan, D., Laskin, M., et al. Vision-language models as a source of rewards. *arXiv preprint arXiv:2312.09187*, 2023.

Bengio, Y., Léonard, N., and Courville, A. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

Bruce, J., Dennis, M., Edwards, A., Parker-Holder, J., Shi, Y., Hughes, E., Lai, M., Mavalankar, A., Steigerwald, R., Apps, C., Aytar, Y., Bechtle, S., Behbahani, F., Chan, S., Heess, N., Gonzalez, L., Osindero, S., Ozair, S., Reed, S., Zhang, J., Zolna, K., Clune, J., de Freitas, N., Singh, S., and Rocktäschel, T. Genie: Generative interactive environments, 2024.

Cetin, E., Tirinzoni, A., Pirotta, M., Lazaric, A., Ollivier, Y., and Touati, A. Simple ingredients for offline reinforcement learning, 2024.

Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling, 2014.

Cui, Y., Niekum, S., Gupta, A., Kumar, V., and Rajeswaran, A. Can foundation models perform zero-shot task specification for robot manipulation?, 2022.

Deng, F., Jang, I., and Ahn, S. Dreamerpro: Reconstruction-free model-based reinforcement learning with prototypical representations. In *International Conference on Machine Learning*, pp. 4956–4975. PMLR, 2022.

Embodiment Collaboration et al. Open x-embodiment: Robotic learning datasets and rt-x models, 2024.

Englert, P., Paraschos, A., Peters, J., and Deisenroth, M. P. Model-based imitation learning by probabilistic trajectory matching. In *2013 IEEE international conference on robotics and automation*, pp. 1922–1927. IEEE, 2013.

Escontrela, A., Adeniji, A., Yan, W., Jain, A., Peng, X. B., Goldberg, K., Lee, Y., Hafner, D., and Abbeel, P. Video prediction models as rewards for reinforcement learning. *Advances in Neural Information Processing Systems*, 36, 2024.

Fan, L., Wang, G., Jiang, Y., Mandlekar, A., Yang, Y., Zhu, H., Tang, A., Huang, D.-A., Zhu, Y., and Anandkumar, A. Minedojo: Building open-ended embodied agents with internet-scale knowledge. *Advances in Neural Information Processing Systems*, 35:18343–18362, 2022.

Fujimoto, S. and Gu, S. S. A minimalist approach to offline reinforcement learning, 2021.

Fujimoto, S., van Hoof, H., and Meger, D. Addressing function approximation error in actor-critic methods, 2018.

Gemini Team et al. Gemini: A family of highly capable multimodal models, 2024.

Graves, A., Fernández, S., Gomez, F., and Schmidhuber, J. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pp. 369–376, 2006.

Gu, A. and Dao, T. Mamba: Linear-time sequence modeling with selective state spaces, 2023.

Ha, D. and Schmidhuber, J. World models. 2018. doi: 10.5281/ZENODO.1207631. URL https://zenodo.org/record/1207631.

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

Hafner, D., Lillicrap, T., Ba, J., and Norouzi, M. Dream to control: Learning behaviors by latent imagination, 2020.

Hafner, D., Pasukonis, J., Ba, J., and Lillicrap, T. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

Klissarov, M., D'Oro, P., Sodhani, S., Raileanu, R., Bacon, P.-L., Vincent, P., Zhang, A., and Henaff, M. Motif: Intrinsic motivation from artificial intelligence feedback, 2023.

Kondratyuk, D., Yu, L., Gu, X., Lezama, J., Huang, J., Hornung, R., Adam, H., Akbari, H., Alon, Y., Birodkar, V., et al. Videopoet: A large language model for zero-shot video generation. *arXiv preprint arXiv:2312.14125*, 2023.

Kostrikov, I., Nair, A., and Levine, S. Offline reinforcement learning with implicit q-learning, 2021.

Liang, W., Zhang, Y., Kwon, Y., Yeung, S., and Zou, J. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning, 2022.

Lifshitz, S., Paster, K., Chan, H., Ba, J., and McIlraith, S. Steve-1: A generative model for text-to-behavior in minecraft, 2024.

Lin, J., Du, Y., Watkins, O., Hafner, D., Abbeel, P., Klein, D., and Dragan, A. Learning to model the world with language, 2023.

Liu, H., Yan, W., Zaharia, M., and Abbeel, P. World model on million-length video and language with blockwise ringattention, 2024.

Ma, Y. J., Liang, W., Wang, G., Huang, D.-A., Bastani, O., Jayaraman, D., Zhu, Y., Fan, L., and Anandkumar, A. Eureka: Human-level reward design via coding large language models, 2024.

Mazzaglia, P., Verbelen, T., Dhoedt, B., Lacoste, A., and Rajeswar, S. Choreographer: Learning and adapting skills in imagination. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PhkWyijGi5b.

OpenAI et al. Gpt-4 technical report, 2024.

R. Bommasani et al. On the opportunities and risks of foundation models, 2022.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. Hierarchical text-conditional image generation with clip latents, 2022.

Reed, S., Zolna, K., Parisotto, E., Colmenarejo, S. G., Novikov, A., Barth-Maron, G., Gimenez, M., Sulsky, Y., Kay, J., Springenberg, J. T., Eccles, T., Bruce, J., Razavi, A., Edwards, A., Heess, N., Chen, Y., Hadsell, R., Vinyals, O., Bordbar, M., and de Freitas, N. A generalist agent, 2022.

Rocamonde, J., Montesinos, V., Nava, E., Perez, E., and Lindner, D. Vision-language models are zero-shot reward models for reinforcement learning. *arXiv preprint arXiv:2310.12921*, 2023.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models, 2022.

Samsami, M. R., Zholus, A., Rajendran, J., and Chandar, S. Mastering memory tasks with world models, 2024.

Sekar, R., Rybkin, O., Daniilidis, K., Abbeel, P., Hafner, D., and Pathak, D. Planning to explore via self-supervised world models, 2020.

Tarasov, D., Kurenkov, V., Nikulin, A., and Kolesnikov, S. Revisiting the minimalist approach to offline reinforcement learning, 2023.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need, 2023.

Wang, G., Xie, Y., Jiang, Y., Mandlekar, A., Xiao, C., Zhu, Y., Fan, L., and Anandkumar, A. Voyager: An open-ended embodied agent with large language models, 2023.

Wang, Y., Li, K., Li, X., Yu, J., He, Y., Chen, G., Pei, B., Zheng, R., Xu, J., Wang, Z., Shi, Y., Jiang, T., Li, S., Zhang, H., Huang, Y., Qiao, Y., Wang, Y., and Wang, L. Internvideo2: Scaling video foundation models for multimodal video understanding, 2024.

Wu, P., Escontrela, A., Hafner, D., Goldberg, K., and Abbeel, P. Daydreamer: World models for physical robot learning, 2022.

Yang, M., Du, Y., Ghasemipour, K., Tompson, J., Kaelbling, L., Schuurmans, D., and Abbeel, P. Learning interactive real-world simulators, 2024.

Yarats, D., Fergus, R., Lazaric, A., and Pinto, L. Mastering visual continuous control: Improved data-augmented reinforcement learning, 2021.

Zhai, X., Mustafa, B., Kolesnikov, A., and Beyer, L. Sigmoid loss for language image pre-training, 2023.

Zhang, Y., Sui, E., and Yeung-Levy, S. Connect, collapse, corrupt: Learning cross-modal tasks with uni-modal data, 2024.

Zhou, Y., Zhang, R., Chen, C., Li, C., Tensmeyer, C., Yu, T., Gu, J., Xu, J., and Sun, T. Lafite: Towards language-free training for text-to-image generation, 2022.

A. Related Work

[Linked to Section 2]

Generative world models for RL. In model-based RL, the optimization of the agent's actions is done efficiently, by rolling out and scoring imaginary trajectories using a (learned) model of the environment's dynamics. In recent years, this paradigm has grown successful thanks to the adoption of generative world models, which learn latent dynamics by self-predicting the agent's inputs (Ha & Schmidhuber, 2018). World models have shown impressive performance in vision-based environments (Hafner et al., 2020), improving our ability to solve complex and open-ended tasks (Hafner et al., 2023). Generative world models have been successfully extended to many applications, such as exploration (Sekar et al., 2020), skill learning (Mazzaglia et al., 2023), solving long-term memory tasks (Samsami et al., 2024), and robotics (Wu et al., 2022).

Recent research has also focused on the question of how to learn world models from large-scale video datasets (Liu et al., 2024; Yang et al., 2024). In (Bruce et al., 2024), they leverage a latent action representation, but their work is mostly focussed on 2D platform videogames or simple robotic actions. In (Escontrela et al., 2024), they use frame-by-frame video prediction as a way to provide rewards for RL. DynaLang (Lin et al., 2023) studies the incorporation of language prediction as part of the world model, to train multimodal world models also from datasets without actions or rewards. The representation in DynaLang is shared in the world model between vision and language, while for GenRL, the world model representation is trained on vision-only data and connected-aligned to the multimodal foundation representation.

Foundations models for RL. Large language models (LLMs) have been used for specifying behaviors using language (Ma et al., 2024; Klissarov et al., 2023; Wang et al., 2023), but this generally assumes the availability of a textual interface with the environment or that observations and/or actions can be translated to the language domain. The adoption of vision-language models (VLMs) reduces these assumptions, as it allows the evaluation of behaviors in the visual space. However. this approach has yet to show robust performance, as it generally requires fine-tuning of the VLM (Baumli et al., 2023; Fan et al., 2022), prompt hacking techniques (Cui et al., 2022) or visual modifications to the environment (Baumli et al., 2023).

Few cases of foundation models for embodied domains have been developed until now. Notable mentions are GATO (Reed et al., 2022), a large-scale behavior cloning agent, trained on 604 tasks. VPT (Baker et al., 2022) a large-scale model trained on Minecraft data, using human-expert labeled trajectories. The model learns strong behavioral priors by behavior cloning which can be fine-tuned using RL. STEVE-1 (Lifshitz et al., 2024) connects VPT's behavioral prior with the MineCLIP model representation (Fan et al., 2022), using the unCLIP approach (Ramesh et al., 2022). RT-X (Embodiment Collaboration et al, 2024) are large-scale trasnformer models trained on expert robotics dataset, sharing a common action space (end-effector pose) across different embodiments.

Vision-language generative modelling. Given the large success of image-language generative models (Rombach et al., 2022), recent efforts in the community have focused on replicating and extending such success to the video domain, where the temporal dimension introduces new challenges, such as temporal consistency and increased computational costs (Kondratyuk et al., 2023; Bar-Tal et al., 2024). Video generative models are similar to world models for RL, with the difference that generation models outputs are typically not conditioned on actions, but rather conditioned on language(Kondratyuk et al., 2023) or on nothing at all (i.e. an unconditional model).

B. Implementation details

[Linked to Section 3]

Actor-critic. Rewards can be maximized over time in imagination in a RL fashion, using actor-critic models of the form:

$$\text{Actor:} \quad \pi_\theta(a_t|s_t), \qquad\qquad \text{Critic:} \quad v_\theta(R_t^\lambda|s_t), \quad \text{where} \quad R_t^\lambda = r_t + \gamma[(1-\lambda v_{t+1}) + \lambda R_{t+1}^\lambda]$$

For the actor-critic, we follow the implementation advances proposed in DreamerV3 (Hafner et al., 2023) (version 1 of the paper, dated January 2023), such as using a two-hot distribution for learning the critic network and scaling returns in the actor loss.

When computing the reward r_{GenRL}, we use the mode of the distribution for the target $s_{t+1}^{\text{task}} \sim p_\psi(s_{t+1}|e_{\text{task}})$ to improve stability.

Hyperparameters. For the hyperparameters, we follow DreamerV3 (Hafner et al., 2023) (version 1 of the paper, dated January 2023). Differences from the default hyperparameters or model size choices are illustrated in Table 2. For instance, a main difference is that we use difference batch sizes/lengths for training the MFWM and the actor-critic as these two stages are now independent from each other.

The connector network uses the same hyperparameters and architecture as the sequential dynamics of the world model. The aligner network employs a small U-Net, with a bottleneck that is half the size of the embedding representation.

Name	**Value**
Multimodal Foundation World Model	
Batch size	48
Sequence length	48
GRU recurrent units	1024
CNN multiplier	48
Dense hidden units	1024
MLP layers	4
Actor-Critic	
Batch size	32
Sequence length	32

Table 2: World model and actor-critic hyperparameters.

C. Tasks

[Linked to Section 4]

We present the list of tasks employed, along with the language prompts used for specifying the task, in Table 3. We introduce a new embodied environment, the *Stickman*, which serves as a humanoid robot that is simpler to control (compared to the one present in the dm_control suite), thanks to a reduced number of joints. Its addition allows studying behaviors that involve upper body movements, rather than focusing on lower body motions. For the newly introduced tasks, the goal can be easily inferred by reading the task's name or its prompt. For the 'flipping' tasks, we consider flips both in forward direction and backward direction, as the VLM struggles to distinguish directions.

The prompts we use have been fine-tuned for the InternVideo2 model (Wang et al., 2024). However, we found that they mostly improved performance for the SigLIP model too (Zhai et al., 2023). One common observation is that these models are generally biased towards human actions. Thus, specifying the embodiment in the prompt is sometimes helpful, e.g. 'spider running fast' or 'running like a quadruped'. Another observation is that for some behaviors the agent can produce

Table 3: Task and prompt used for each task

Task	Prompt	Specialized agent score	Random agent score
quadruped run	spider running fast	930	10
quadruped walk	spider walking fast	960	10
quadruped stand	spider standing	990	15
quadruped jump	spider jumping	875	15
quadruped two legs	on two legs	875	14
quadruped lie down	lying down	965	750
cheetah run	running like a quadruped	890	9
cheetah standing	standing like a human	930	5
cheetah lying down	lying down	920	430
stickman walk	robot walk fast clean	960	35
stickman run	robot run fast clean	830	25
stickman stand	standing	970	70
stickman flipping	doing flips	790	45
stickman one foot	stand on one foot	865	20
stickman high kick	stand up and kick	920	55
stickman lying down	lying down horizontally	965	380
stickman sit knees	praying	966	40
stickman lunge pose	lunge pose	950	100
stickman headstand	headstand	955	180
stickman boxing	punch	920	80
stickman hands up	standing with the hands up	830	5
walker walk	walk fast clean	960	45
walker run	run fast clean	770	30
walker stand	standing up straight	970	150
walker flipping	doing backflips	720	20
walker one foot	stand on one foot	955	20
walker high kick	stand up and kick	960	25
walker lying down	lying down horizontally	975	170
walker sit knees	praying	945	100
walker lunge pose	lunge pose	945	150
kitchen microwave	opening the microwave fully open	1	0
kitchen light	activate the light	1	0
kitchen burner	the burner becomes red	1	0
kitchen slide	slide cabinet above the knobs	1	0

very different styles, e.g. the agent can be walking in a slow or fast way, or in a more or less composed manner. Specifying words like ‘fast’ or ‘clean’ helps clarifying the modality of the expected motion.

D. Experiments settings

[Linked to Section 4]

Baselines. In order to implement performant offline RL baselines we adopt the findings of (Tarasov et al., 2023) and (Cetin et al., 2024), adopting larger deeper networks and layer normalization.

Inputs are 64x64x3 RGB images. We use a frame stack of 3. The encoder architecture is adapted from the DrQ-v2 encoder (Yarats et al., 2021). We did find augmentations on the images, e.g. random shifts, to hurt performance.

Offline RL. For each task, training model-free agents (IQL, TD3, TD3+BC) requires re-training the full agent (visual encoder, actor, critic) on the entire dataset, from scratch, while training model-based agents (GenRL) requires training the model once for each domain and then training an actor-critic for each task. Moreover, for training the actor-critic in GenRL, we only use 50k gradient steps, as the policy converges significantly faster than for the other methods. We employ standard hyperparameters from the original papers.

Compute resources. We use a cluster of V100 with 16GB of VRAM for all our main experiments. To enable efficient training, image and video embeddings from the VLM are computed in advance and stored with the datasets. Training the MFWM for 500k gradient steps takes ~ 5 days. After pre-training the MFWM, training the actor-critic for a prompt for 50k gradient steps takes less than 5 hours. In data-free mode, it takes less than 3 hours. In both cases, convergence normally arrives after 10k gradient steps, but we keep training. Model-free baselines take around 7 hours to train for 500k gradient steps.

Datasets composition. We present the datasets’ composition in Table 4.

Table 4: Datasets composition.

Domain	Count	Subset	Subcount
walker	2.5M	walker run	500k
		walker walk	500k
		walker stand	500k
		walker expl	1M
cheetah	1.8M	cheetah run	1M
		cheetah expl	820k
quadruped	2.5M	quadruped expl	1M
		quadruped run	500k
		quadruped stand	500k
		quadruped walk	500k
kitchen	3.6M	kitchen slide	700k
		kitchen light	700k
		kitchen bottom burner	700k
		kitchen microwave	700k
		kitchen expl	800k
stickman	2.5M	stickman stand	500k
		stickman walk	500k
		stickman expl	1M
		stickman run	500k
minecraft	4M	-	-

E. Behavior Temporal Alignment

[Linked to Section 3]

One issue with trajectory matching is that it assumes that the distribution of states visited by the agent starts from the same state as the target distribution. However, the initial state generated by the connector may differ from the initial state where the policy is currently in. For example, consider the Stickman agent on the right side of Figure 1. If the agent is lying on the ground and tasked to run, the number of steps to get up and reach running states may surpass the temporal span recognized by the VLM (e.g. typically 4, 8, or 16 frames), causing disalignment in the reward.

To address this initial condition alignment issue, we propose a *best matching trajectory* technique, inspired by best path decoding in speech recognition (Graves et al., 2006). Our technique involves two steps:

1. We compare the first b states of the target trajectory with b states obtained from the trajectories imagined by the agent by sliding along the time axis. This allows one to find at which timestep t_a the trajectories are best aligned (the comparison provides the highest reward).
2. We align the temporal sequences in the two possible contexts: (a) if a state from the agent sequence comes before t_a, the reward uses the target sequence's initial state; and (b) if the state comes k steps after t_a, it's compared to the s_{t+k} state from the target sequence.

In all experiments, we fix $b = 8$ (number of frames of the VLM we use (Wang et al., 2024)), which we found to strike a good compromise between comparing only the initial state ($b = 1$) and performing no alignment ($b =$ imagination horizon). An ablation study can be found in Appendix F.

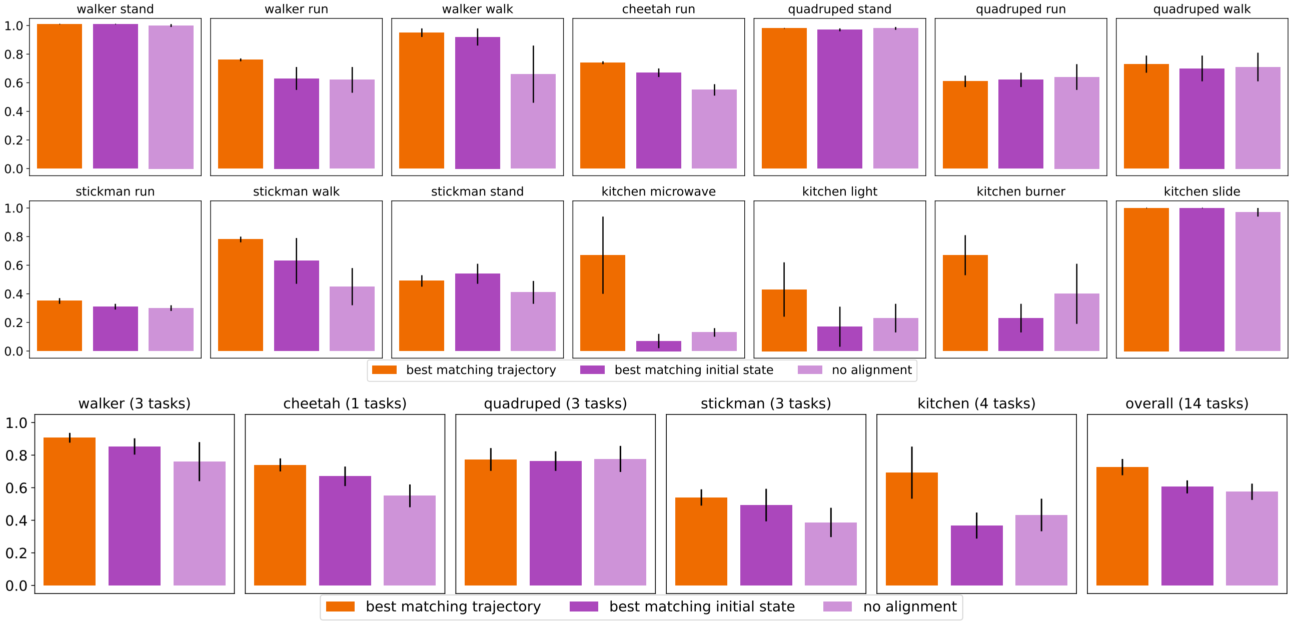


Figure 6: *Temporal alignment ablation.* We analyze the impact of temporal alignment in our proposed RL objective for matching sequential targets. Results averaged over 3 seeds.

F. Additional experiments

[Linked to Section 4]

Table 5: Offline RL from language prompts on tasks that are included in the agent's training dataset. Scores are episodic rewards averaged over 5 seeds (± standard error) rescaled using min-max scaling with $(\min = \text{random}, \max = \text{expert})$.

	Image-language reward			Video-language reward			Ours
	IQL	TD3+BC	TD3	IQL	TD3+BC	TD3	GenRL
walker stand	0.68 ± 0.03	0.97 ± 0.01	0.92 ± 0.06	0.72 ± 0.05	0.59 ± 0.05	**1.0 ± 0.0**	**1.01 ± 0.0**
walker run	0.24 ± 0.03	0.27 ± 0.02	0.11 ± 0.03	0.26 ± 0.02	0.25 ± 0.02	0.35 ± 0.01	**0.76 ± 0.01**
walker walk	0.37 ± 0.04	0.25 ± 0.03	0.15 ± 0.0	0.43 ± 0.04	0.39 ± 0.01	0.86 ± 0.04	**0.95 ± 0.03**
cheetah run	0.37 ± 0.07	-0.01 ± 0.0	-0.01 ± 0.0	0.15 ± 0.03	-0.01 ± 0.0	0.34 ± 0.01	**0.74 ± 0.01**
quadruped stand	0.32 ± 0.06	0.4 ± 0.06	0.62 ± 0.04	0.39 ± 0.04	0.41 ± 0.13	0.73 ± 0.06	**0.98 ± 0.0**
quadruped run	0.3 ± 0.03	0.35 ± 0.01	0.25 ± 0.02	0.36 ± 0.04	0.35 ± 0.04	0.23 ± 0.02	**0.61 ± 0.04**
quadruped walk	0.16 ± 0.02	0.18 ± 0.02	0.37 ± 0.02	0.19 ± 0.04	0.16 ± 0.04	0.37 ± 0.02	**0.73 ± 0.06**
stickman run	0.2 ± 0.02	0.25 ± 0.02	0.03 ± 0.0	0.24 ± 0.02	0.18 ± 0.03	0.21 ± 0.0	**0.35 ± 0.02**
stickman walk	0.43 ± 0.07	0.52 ± 0.05	0.18 ± 0.01	0.47 ± 0.02	0.45 ± 0.08	0.42 ± 0.03	**0.78 ± 0.02**
stickman stand	0.52 ± 0.05	**0.68 ± 0.04**	0.56 ± 0.04	0.56 ± 0.06	0.47 ± 0.02	0.61 ± 0.03	0.49 ± 0.04
kitchen microwave	0.14 ± 0.13	0.0 ± 0.0	0.0 ± 0.0	0.02 ± 0.02	0.0 ± 0.0	0.46 ± 0.18	**0.67 ± 0.27**
kitchen light	0.18 ± 0.12	1.0 ± 0.0	**0.66 ± 0.07**	0.04 ± 0.02	0.0 ± 0.0	0.0 ± 0.0	0.43 ± 0.19
kitchen burner	0.14 ± 0.07	0.43 ± 0.14	0.02 ± 0.02	0.02 ± 0.02	0.0 ± 0.0	0.28 ± 0.08	**0.67 ± 0.14**
kitchen slide	0.16 ± 0.1	0.0 ± 0.0	0.0 ± 0.0	0.06 ± 0.04	0.0 ± 0.0	**1.0 ± 0.0**	**1.0 ± 0.0**
overall	0.30 ± 0.04	0.38 ± 0.02	0.28 ± 0.02	0.28 ± 0.02	0.23 ± 0.02	0.49 ± 0.04	**0.73 ± 0.05**

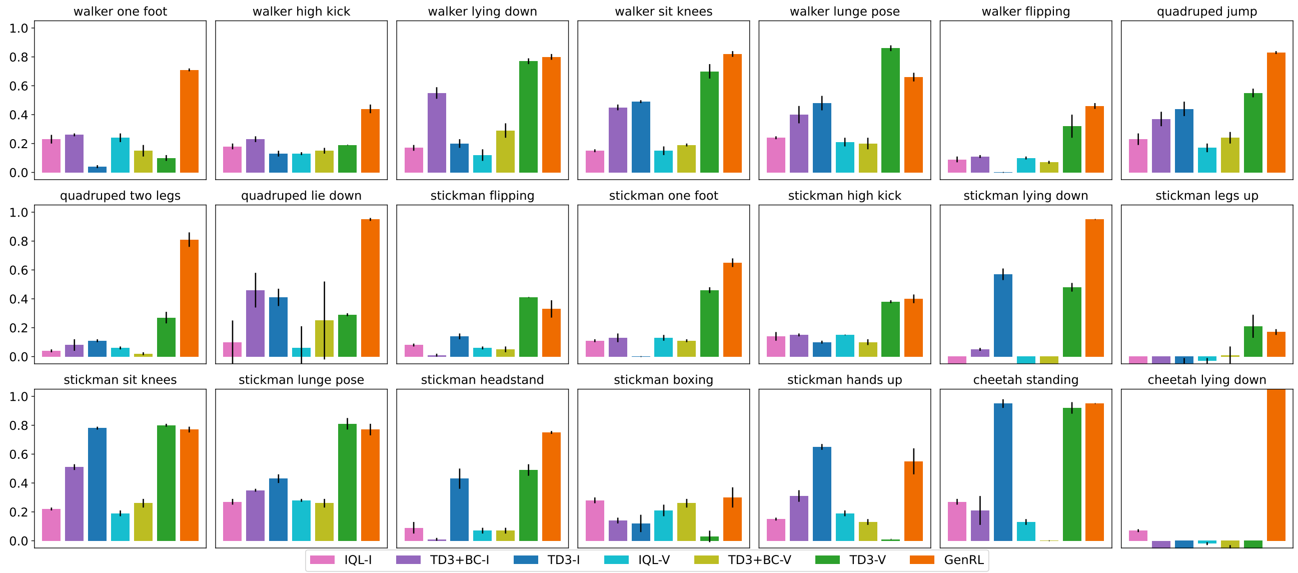


Figure 7: *Multi-task generalization detailed results.* Results averaged over 5 seeds.

F.1. Initial states distribution

Uniform sampling from the latent space of the world model often results in meaningless latent states. Additionally, the sequential dynamics model of the MFWM, using a GRU, requires some 'warmup' steps to discern dynamic environmental attributes, such as velocities.

To address these issues we perform two operations. First, we combine uniformly sampled states from the discrete latent spaces with states generated by randomly sampling the connector model, as sequences generated by the connector tend to have a more coherent structure than random uniform samples. Second, we perform a rollout of five steps using a mix of actions from the trained policy and random actions. This leads to a varied distribution of states, containing dynamic information, which we use as the initial states for the learning in imagination process.

In Figure 8, we show detailed results for data-free learning and we ablate the choice of using random samples from the connector-aligner to improve randomly sampled initial states. The use of states from the connector model enhances average scores and reduces variance, especially noticeable in the cheetah domain.

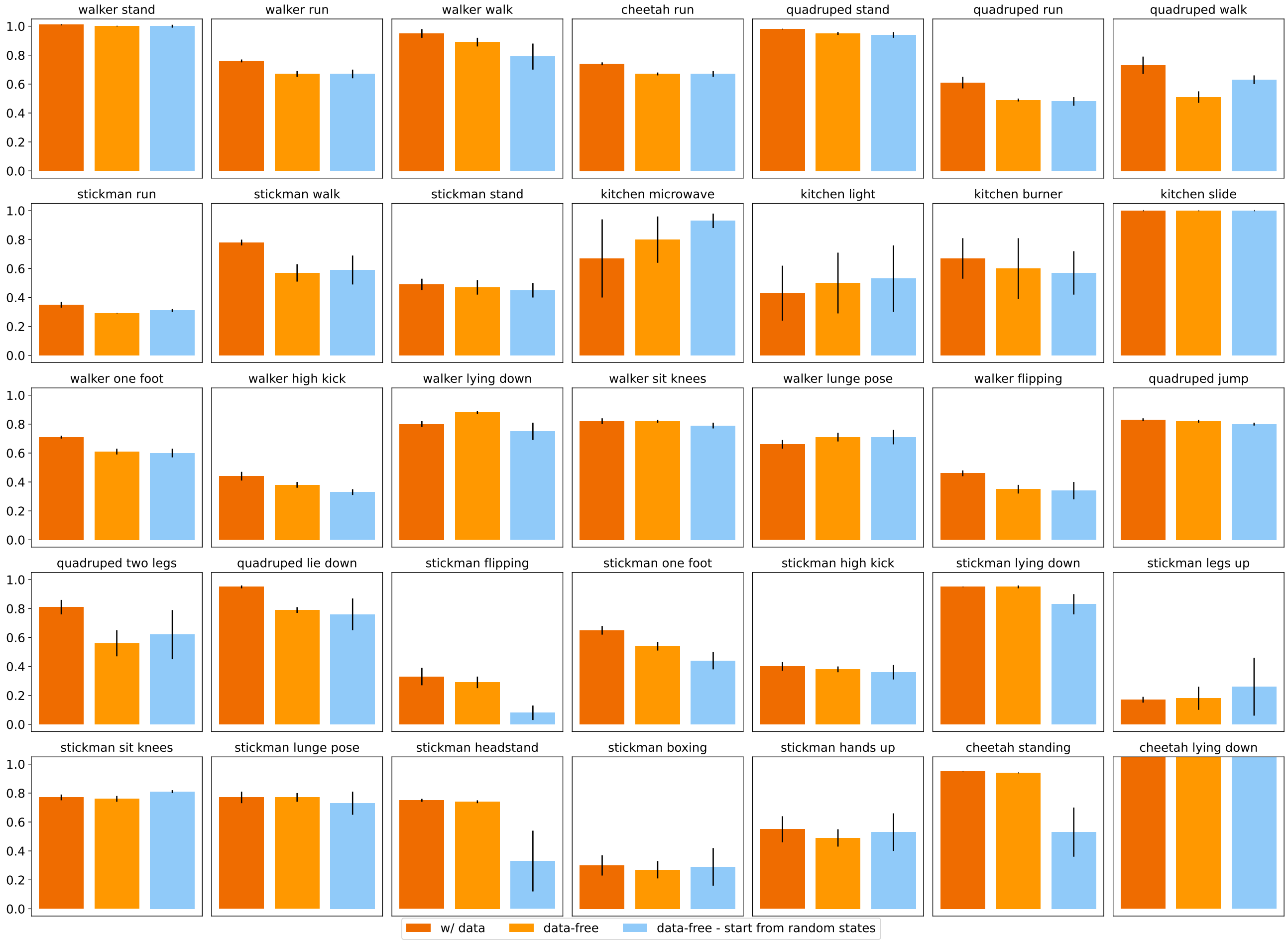


Figure 8: *Data-free RL detailed results*. Results averaged over 5 seeds.

F.2. Training data distribution

As demonstrated in Sections 4.1 and 4.2, after training on a large dataset, a GenRL agent can adapt to multiple new tasks without additional data. The nature of the training data, detailed in Appendix C, combines exploration and task-specific data. To identify critical data types for GenRL, we trained different MFWMs on various dataset subsets. Then, we employ data-free RL to train task behaviors, with analyses over subsets of the walker dataset provided in Figure 9, where ‘all’ reports the data-free performance when training on the full dataset.

The results confirm that a diverse data distribution is crucial for task success, with the best performance achieved by using the complete dataset, followed by the varied exploration data. Task-specific data effectiveness depends on task complexity, for instance, ’run’ data proves more generalizable than ’walk’ or ’stand’ data across tasks. Crucially, ’stand’ data, which shows minimal variation, limits learning for a general agent but can still manage simpler tasks like ’lying down’ and ’sitting on knees’.

Moving forward with training foundation models in RL, it will be essential to develop methods that extract multiple behaviors from unstructured data and accurately handle complex behaviors from large datasets. Thus, the ability of GenRL to primarily leverage unstructured data is a significant advantage for scalability.

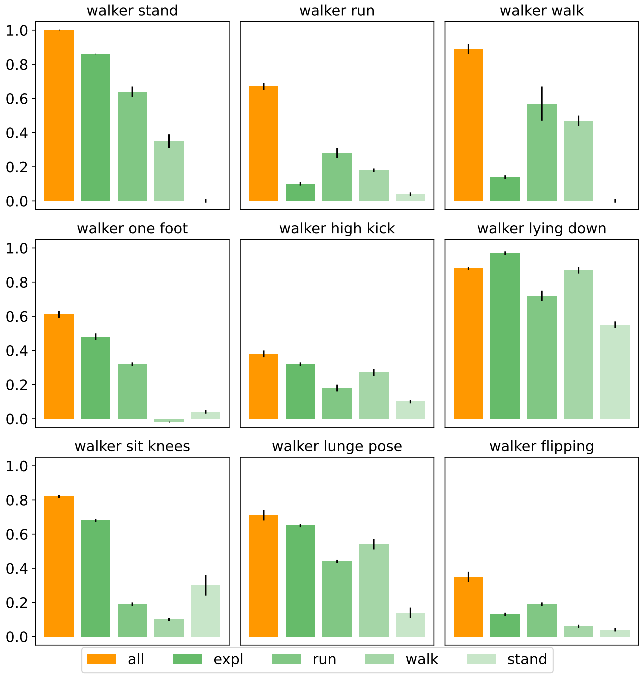


Figure 9: *Training data distribution detailed results*. Results averaged over 3 seeds.

F.3. Scaling to complex observations.

Generalist embodied agents should be able to scale to open-ended learning settings. Using GenRL, we explored this by training an agent in the Minecraft environment using a small dataset collected by a DreamerV3 agent (Hafner et al., 2023). The primary challenge we found was the model's difficulty in reconstructing complex observations in this open-ended environment.

Reconstructing complex observations is a common issue with world models (Deng et al., 2022). To overcome this limitation, while keeping the method unaltered, we attempted to scale up the number of parameters of MFWM. Qualitative reconstruction results are presented in Figure 10. We observe that the agent is able to identify different biomes from language, even with the smaller size of the model. However, the reconstructions are significantly blurrier compared to the other environments we analyzed (e.g. Figure 5). When using a larger model, the reconstructions gain some details but the results still highlight the difficulty of the model in providing accurate targets from prompts.

While this might not be an issue for simple high-level tasks, e.g. 'navigate to a beach', unclear targets might make it difficult to perform more precise actions, e.g. 'attack a zombie'. Future research should aim to address this issue, for instance, by improving our simple GRU-based architecture, leveraging transformers or diffusion models to improve the quality of the representation (Kondratyuk et al., 2023; Bar-Tal et al., 2024).

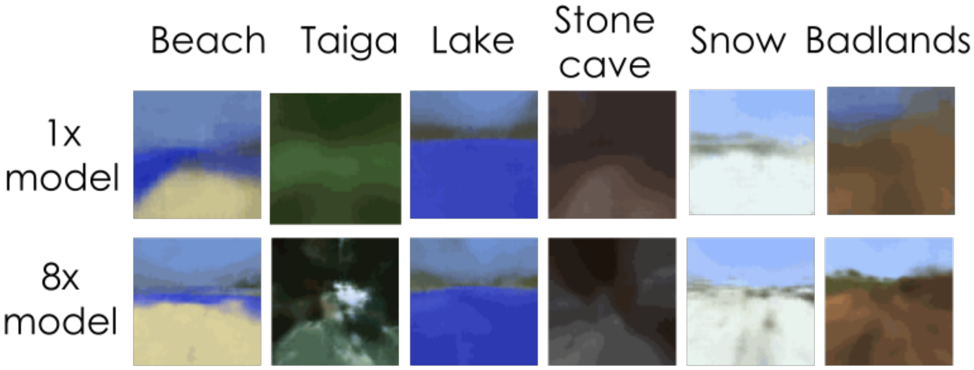


Figure 10: Decoded language prompts in Minecraft

G. Extended Discussion

We introduced GenRL, a world-model based approach for grounding vision-language prompts into embodied domains and learning the corresponding behaviors in imagination. The multimodal foundation world models of GenRL can be trained using unimodal data, overcoming the lack of multimodal data in embodied RL domains.

By employing data-free learning, after pre-training, agents can master new tasks without data, often converging within only 30 minutes of GPU training. As we scale up foundation models for behavior learning, the ability to learn data-free will become crucial. Although very large datasets will be employed to train new foundation models, GenRL adapts well without direct access to original data, offering flexibility where data may be proprietary, licensed or unavailable.

Limitations. Despite its strengths, GenRL presents some limitations, largely due to inherent weaknesses in its components. From the VLMs, GenRL inherits the issue related to the multimodality gap (Liang et al., 2022; Zhang et al., 2024) and the reliance on prompt tuning. We proposed a connection-alignment mechanism to mitigate the former. For the latter, we presented an explainable framework, which facilitates prompt tuning by allowing decoding of the latent targets corresponding to the prompts. From the world model, GenRL inherits a dependency on reconstructions, which offers advantages such as explainability but also drawbacks, such as failure modes with complex observations.

Future work. As we strive to develop foundation models for generalist embodied agents, our framework opens up numerous research opportunities. One such possibility is to learn multiple behaviors and have another module, e.g. an LLM, compose them to solve long-horizon tasks. Another promising area of research is investigating the temporal flexibility of the GenRL framework. We witnessed that for static tasks, greater temporal awareness could enhance performance. This concept could also apply to actions that extend beyond the time comprehension of the VLM. Developing general solutions to these challenges could lead to significant advancements in the framework.