From World Models to World Action Models: A Concise Tutorial for Robotics

Xiaoxiong Zhang [1] , **Xiong Zeng** [1] , and **Wei Zhang** [1 2]

Abstract

World models are increasingly used in embodied intelligence and generative simulation, yet their scope remains ambiguous across communities. This tutorial presents a design-space view of world models as action-conditioned predictive models that estimate the future evolution of task-relevant observations or states. We categorize existing methods into observation-space and state-space world models, comparing their trade-offs in visual fidelity, spatial structure, physical interpretability, and control usability. We further introduce world action models, which connect predicted futures with executable robot actions, and summarize four representative paradigms: imagine-then-execute, video-feature-conditioned action prediction, joint video-action modeling, and auxiliary video prediction for policy learning. The goal of this tutorial is to clarify the conceptual scope of world (action) models and provide a structured taxonomy for embodied prediction and control.

Project page: World (Action) Models.

1. Architectural View of World (Action) Models in Robotics

Predicting how the world evolves under actions is a central capability for embodied artificial intelligence (AI) and generative simulation. In these scenarios, the predictive models can support controller design, planning, decision making, and policy learning. Recently, the term *world (action) model* has been used to describe a broad range of methods, including latent dynamics models, video prediction models, physics-informed simulators, and action-conditioned generative models. However, these methods differ substantially in what they model, what they predict, and how their predictions are used. This motivates a unified definition and taxonomy of world (action) models.

[1]School of Automation and Intelligent Manufacturing, Southern University of Science and Technology, Shenzhen, China [2]LimX Dynamics, Shenzhen, China. Correspondence to: Xiaoxiong Zhang <12433017@mail.sustech.edu.cn>.

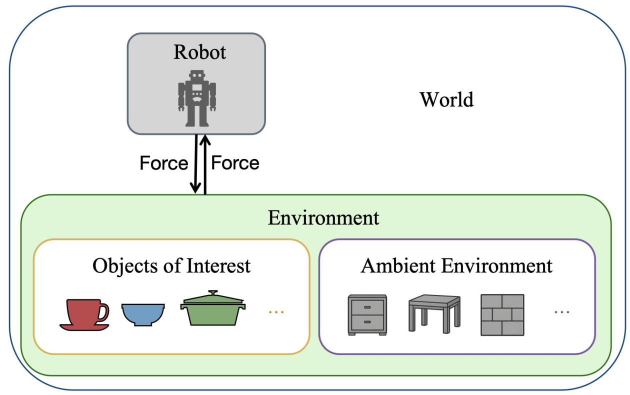

Figure 1. Illustration of the components of a world.

1.1. World

We define a *world* as the set of task-relevant entities, including both the *robot* and its *environment*. The environment contains the *objects of interest* and the *ambient environment*, as shown in Fig. 1.

1.2. Embodied AI Task and Policy

We define the *world configuration* as the configuration of both the robot and all objects inside the environment. An *embodied AI task* is to design a policy to transform the initial world configuration into a target configuration by controlling the robot. Different embodied AI tasks correspond to different instantiations of the world.

Fig. 2 provides two examples of task-specific worlds. In humanoid locomotion on flat ground, the world consists of a humanoid robot and a ground surface, and the goal is to induce stable and feasible motion configurations. In robotic table-cleaning tasks after dinner, the world includes the robot and all relevant objects such as dishes, a table, and a household environment, and the objective is to rearrange objects into acceptable configurations (e.g., dishes placed in a sink).

Fig. 3 illustrates that, in an embodied AI task, the *policy* receives a language instruction l and the current observation

Figure 2. Embodied AI tasks: Humanoid locomotion and robotic table-cleaning.

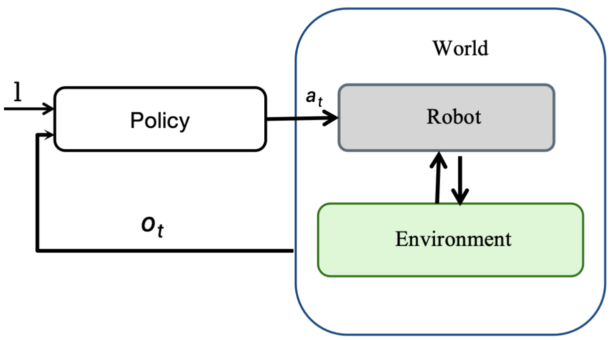

Figure 3. A language-conditioned closed-loop policy framework.

o_t from the world, and then outputs an action a_t to the robot. The policy might be a proportional-integral-derivative (PID) controller, a model predictive controller (MPC), a vision-language-action (VLA) model, or a world action model (WAM).

1.3. World Models and World Action Models

For a specified world, a *world model* is a model to predict how its future observation o_{t+1} or state x_{t+1} evolves under action a_t, typically conditioned $o_{0:t}$, as illustrated in Fig. 4.

The model may take different forms, including symbolic dynamics equations, neural network dynamics models, or diffusion-based video predictors, as shown in Fig. 5. The observation o_t could be an RGB or RGB-D image, a point cloud, or a robot proprioceptive state. The action a_t might range from a concrete agent action to an abstract language-level instruction or a camera pose for observing the environment. The state x_t may represent object poses, keypoints, a latent state, or other task-relevant variables. We discuss world models in more detail in Sec. 2.

World action model is a policy in the embodied AI task in

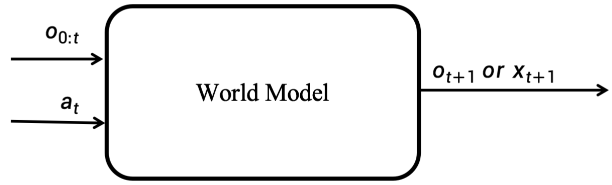

Figure 4. A world model predicts future observations or states from observation history and action.

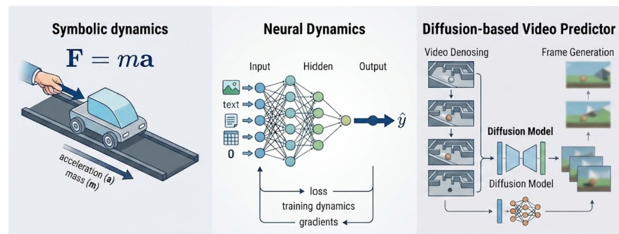

Figure 5. Examples of world models.

Fig. 3. It extends world models by explicitly associating predicted future observations or states with actions. More details about world action model can be found in Sec. 3.

2. A Design-Space View of World Models

While Sec. 1 defines a world model abstractly as an action-conditioned predictive model, existing world models can be broadly categorized according to the space in which prediction is performed. A one-step world model can be written as

$$y_{t+1} \sim p_\theta(\cdot \mid o_{0:t}, a_t),$$

where p_θ denotes a parameterized predictive model, y_{t+1} denotes the prediction target, which could be the future observation o_{t+1} or the future state x_{t+1}. When $y_{t+1} = o_{t+1}$, we refer to the model as an *observation-space* world model. Similarly, when $y_{t+1} = x_{t+1}$, we call it a *state-space world model*. More generally, the prediction target can be a future trajectory, such as $y_{t+1:t+H}$.

2.1. Observation-space World Models

An observation-space world model predicts future observations directly in the observation space under a given action. Given an observation history $o_{0:t}$ and an action a_t, the model estimates the predictive distribution of the next observation:

$$o_{t+1} \sim p_\theta(\cdot \mid o_{0:t}, a_t),$$

where p_θ is typically parameterized by a neural network. Under this formulation, the design space of observation-space world models can be largely characterized by two axes: the type of observation being predicted and the level

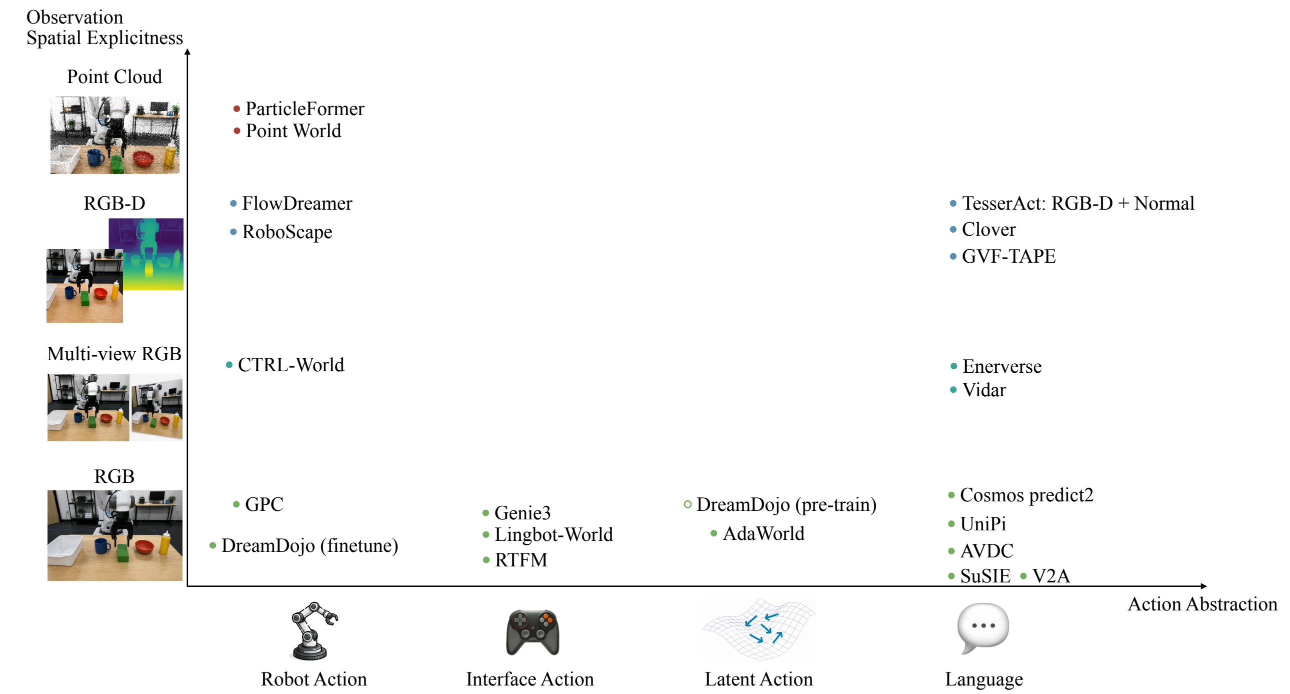

Figure 6. Design space of observation-space world models. The vertical axis denotes the spatial explicitness of the observation, ranging from RGB images to multi-view RGB, RGB-D, and point clouds. The horizontal axis denotes the abstraction level of the action conditioning, ranging from low-level robot actions to interface actions, latent actions, and language instructions. Different choices along these two axes lead to different trade-offs between spatial understanding, data availability, and downstream usability.

of abstraction of the action conditioning, as illustrated in Fig. 6.

Classification by observation spatial explicitness. On the observation side, RGB images are the most common representation because they are easy to collect, require inexpensive and widely available sensors, and are abundant in Internet-scale video data (Goo; Black et al., 2023; Du et al., 2023; Gao et al., 2025; 2026; Ko et al., 2023; Luo & Du, 2025; NVIDIA et al., 2026; Qi et al., 2026; Team et al., 2026; World Labs Team, 2025). However, downstream decision-making tasks, especially robotic manipulation, often require the model to capture the spatial structure of the physical scene. This motivates the use of observations with increasing spatial explicitness, including multi-view RGB images (Feng et al., 2025; Guo et al., 2026; Huang et al., 2025a), RGB-D images (Bu et al., 2024; Guo et al., 2025; Shang et al., 2025; Zhang et al., 2025; Zhen et al., 2025), and point clouds (Huang et al., 2025b; 2026). These representations provide progressively richer geometric information, but they also introduce a trade-off: spatially explicit data are much less available at scale than ordinary RGB videos. A common compromise is therefore to exploit large-scale RGB video data while enhancing their spatial information through monocular vision models, for example by annotating videos with estimated depth to form pseudo-RGB-D data (Guo et al., 2025; Shang et al., 2025; Zhang et al., 2025; Zhen et al., 2025).

Classification by action abstraction. On the action side, the choice of action representation is closely tied to the intended role of the world model. While the observation type determines what form of future is predicted, the action type largely determines how the model can be used. A more concrete action representation usually makes the model suitable for control and simulation, whereas a more abstract action representation often shifts the model toward pretraining or planning.

- At the most concrete level, actions correspond to low-level robot or agent commands, such as joint commands, end-effector actions, or continuous control signals (Gao et al., 2026; Guo et al., 2025; 2026; Huang et al., 2025b; 2026; Qi et al., 2026; Shang et al., 2025). Since these actions are directly grounded in the embodiment of the agent, the resulting world model can predict the physical consequence of executing a specific control command. Such models are therefore commonly used as learned simulators for model-predictive control (Guo et al., 2025; Huang et al., 2025b; 2026;

Qi et al., 2026), policy evaluation (Guo et al., 2026; Shang et al., 2025), or synthetic data generation (Guo et al., 2026; Shang et al., 2025).

- At the next level, interface actions provide a controllable way to interact with the visual world without specifying the low-level motor commands of a particular embodiment. For example, given an initial image or scene context, the model may take user-level control signals, camera-control commands, or viewpoint-control inputs, and then synthesize the corresponding future observations (Goo; Team et al., 2026; World Labs Team, 2025). The purpose of this type of world model is not to predict robot dynamics, but to turn a static or partially observed scene into an interactive visual environment. It therefore serves as an interface-driven visual simulator, where users or agents can control how the scene is observed and obtain temporally consistent observations under different interactions.
- Another line of work introduces latent actions, which are learned from unlabeled videos through reconstruction objectives with an information bottleneck (Gao et al., 2025; 2026). Here, the action is not provided by the dataset, but inferred as a compact variable that explains the visual transition between frames. This design is mainly motivated by pretraining: by replacing missing action labels with learned latent actions, the world model can exploit large-scale action-free video data and acquire general dynamics priors before being adapted to downstream tasks.
- At the most abstract level, actions can be represented as language instructions. Language-conditioned world models are less grounded in precise low-level control, but they are useful for high-level visual planning. Given a task description, such models generate possible future observations that describe how the task may unfold or be completed, thereby providing visual guidance for downstream agents (Black et al., 2023; Bu et al., 2024; Du et al., 2023; Feng et al., 2025; Huang et al., 2025a; Ko et al., 2023; Luo & Du, 2025; NVIDIA et al., 2026; Zhang et al., 2025; Zhen et al., 2025).

These action representations form a hierarchy ranging from low-level motor control to high-level semantic instructions.

2.2. State-space World Models

Predicting future observations directly in the observation space is a natural formulation, but it is often difficult to learn because raw observations are high-dimensional and contain many task-irrelevant variations, such as background changes, illumination, texture, or sensor noise. As a result, modeling dynamics directly in the observation space may force the model to explain unnecessary visual details rather than the task-relevant evolution of the world. State-space world models address this issue by first abstracting observations into a more compact and structured state representation, and then predicting the future evolution in this state space:

$$x_{t+1} \sim p_\theta(\cdot \mid o_{0:t}, a_t),$$

where x_{t+1} denotes the future state of interest, which may be a latent vector, a set of tracked points, a collection of symbolic predicates, or physical state variables. Different from observation-space world models, the transition model p_θ is not necessarily restricted to a neural network. Depending on the choice of state representation, the transition model may be implemented as a neural network, a symbolic or neural-symbolic operator model, or a physics-based simulator. By predicting structured states instead of raw observations, state-space world models reduce the complexity of dynamics learning and focus the prediction on task-relevant or physically meaningful aspects of the world. As illustrated in Fig. 7, existing state-space world models can be organized according to the type of state representation they use.

Latent state model. A major class of state-space world models uses latent states, where observations are encoded into latent vectors by neural networks (Assran et al., 2025; Goswami et al., 2025; Hafner et al., 2023; Hansen et al., 2024; Huang et al., 2025c; Jeong et al., 2025; Maes et al., 2025; Wu et al., 2022; Yin et al., 2025; Zhou et al., 2024). These latent states can be obtained in different ways. Some methods learn the latent representation directly from the target domain, for example through reconstruction-based objectives (Hafner et al., 2023; Wu et al., 2022) or through predictive objectives that model future latent states without reconstructing pixels (Assran et al., 2025; Goswami et al., 2025; Hansen et al., 2024; Maes et al., 2025). Such representations are usually adapted to the specific environment, task distribution, and control setting. Other methods use pretrained visual or vision-language models to extract latent features as state representations (Huang et al., 2025c; Jeong et al., 2025; Yin et al., 2025; Zhou et al., 2024). Since these encoders are trained on large-scale Internet data, the resulting states often provide more general semantic and visual abstractions. Latent-state world models are commonly paired with robot or agent actions and are often used as compact learned simulators for reinforcement learning, or model-predictive control.

Point track model. Another line of work represents the state as point tracks (Wang et al., 2025b; Wen et al., 2024; Xu et al., 2024; Zhi et al., 2025). These points can be two-dimensional (Wang et al., 2025b; Wen et al., 2024; Xu et al., 2024) or three-dimensional (Zhi et al., 2025), randomly sampled (Wen et al., 2024) or semantically selected (Wang

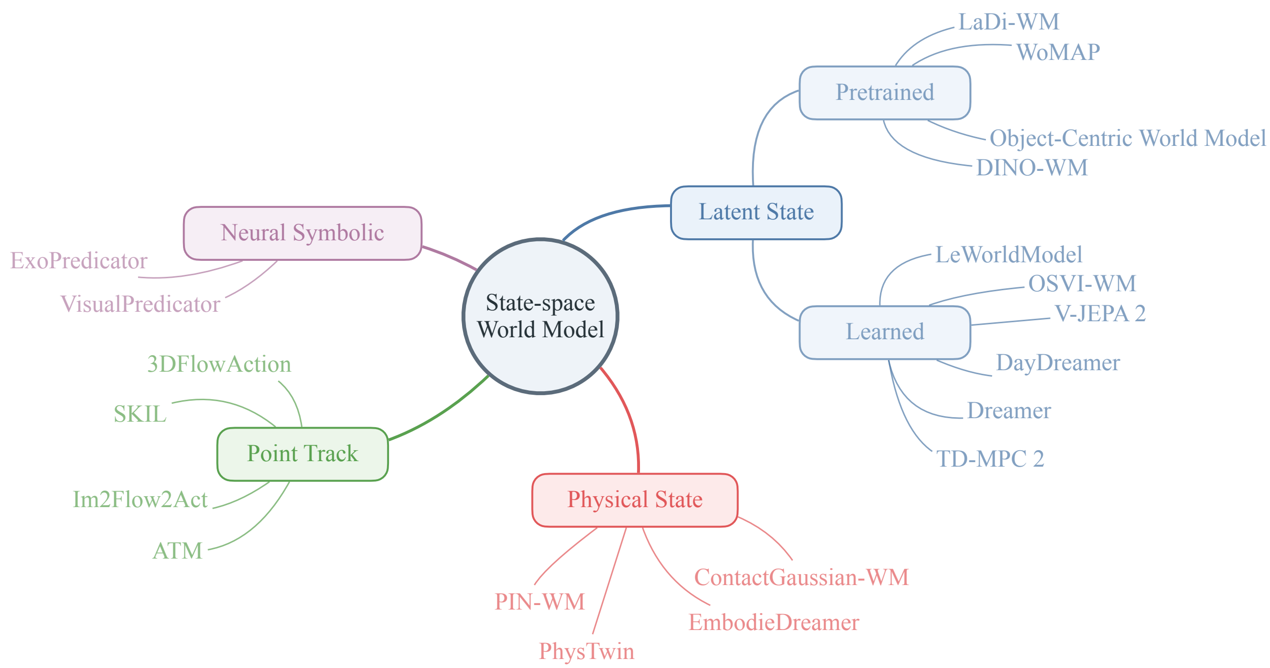

Figure 7. Design space of state-space world models. Instead of predicting future observations directly in the raw observation space, state-space world models abstract observations into structured state representations and model their future evolution under actions. Representative state choices include latent states, point tracks, neural-symbolic predicates, and physical states. Different state representations provide different trade-offs between compactness, semantic structure, physical interpretability, scalability, and downstream usability.

et al., 2025b; Xu et al., 2024; Zhi et al., 2025), and their trajectories describe how task-relevant parts of the scene move over time. Compared with raw image prediction, point-track prediction removes much of the irrelevant background variation while preserving explicit motion information in the scene. This makes the future representation more compact and structured, without requiring the model to generate full visual observations. However, the choice of points also introduces an additional structural prior: although this prior can simplify learning, it may reduce scalability when the relevant scene elements are hard to define or vary significantly across tasks. In many applications, point-track world models are conditioned on language instructions and predict how key points in the scene should move in order to accomplish the task. The predicted tracks can then be used as reference trajectories or intermediate goals for a downstream controller.

Neural-symbolic model. Neural-symbolic world models use a set of grounded predicates as the state representation (Liang et al., 2024b; 2025b). A neural-symbolic predicate is a symbolic Boolean fact whose truth value is grounded in raw perception by a neural model. This representation maps continuous visual observations into a compact set of logical facts that can support reasoning and planning. In this setting, actions are usually high-level skills rather than low-level motor commands. The transition model describes the preconditions and effects of these skills over the predicate state, similar in spirit to symbolic planning models such as PDDL, but with predicates grounded in perception. The goal of this type of world model is to learn a compact symbolic abstraction for long-horizon planning. By representing skills through their preconditions and effects, neural-symbolic world models support compositional reasoning, efficient search, and interpretable planning over high-level task structures.

Physical state model. A further class of state-space world models is based on physical states and classical mechanics (Jiang et al., 2025; Li et al., 2025b; Wang et al., 2025a; 2026). Instead of learning arbitrary latent dynamics from data, these methods explicitly represent the world using physical variables such as object poses, velocities, contact states, masses, friction coefficients, or other dynamics parameters. This formulation incorporates strong human priors about how the physical world evolves, and is closely related to the modeling assumptions used in physics simulators. A typical pipeline first reconstructs the target scene in 3D, then aligns a simulated environment with the real scene by replaying real interaction trajectories and optimizing dynamics and rendering parameters. Once the simulator is aligned, future states can be predicted by solving physics-based transition equations conditioned on the current physical state and the agent action. Such world models are especially useful for model-predictive control, reinforcement learning, and synthetic data generation, where accurate physical rollouts are more important than directly generating raw visual observations.

3. World Action Models

The previous section categorizes world models according to their prediction space. We now focus on a practically important subclass: language-conditioned observation-space world models that generate visual futures. In embodied decision making, prediction alone is insufficient; the robot must also infer executable actions that can realize the predicted future. This motivates world action models, which couple visual future prediction with action generation:

$$(o_{t+1:t+H}, a_{t:t+H-1}) \sim p_\psi(\cdot \mid o_t, l).$$

Existing approaches differ in how this coupling is implemented, leading to four representative paradigms summarized in Fig. 8.

Imagine-then-execute. The most direct paradigm is a two-stage *imagine-then-execute* formulation (Black et al., 2023; Du et al., 2023; Feng et al., 2025; Ko et al., 2023; Liang et al., 2024a; Patel et al., 2026; Wen et al., 2024; Zhang et al., 2025). In the first stage, a video prediction model generates visual subgoals:

$$o_{t+1:t+H} \sim p_\theta(\cdot \mid o_t, l).$$

In the second stage, a separate inverse dynamics model or goal-conditioned policy, denoted by the parameterized distribution q_ϕ, maps the current observation and a predicted visual subgoal into robot actions,

$$a_t \sim q_\phi(\cdot \mid o_t, o_{t+1}).$$

The two modules together form a world action model: the video model imagines a possible future, and the inverse dynamics model grounds this future into executable actions. This decomposition has several practical advantages. Since the visual planner and the action grounding model are separated, they can be trained on different data sources: the video model can benefit from large-scale visual data, while the inverse dynamics model can be trained on robot trajectories with action labels. The modular design also allows additional structure to be injected into the action grounding process. For example, one can estimate intermediate quantities such as end-effector poses (Liang et al., 2024a; Zhang et al., 2025), object poses (Patel et al., 2026), or dense optical flow (Ko et al., 2023) from the generated video, and then use these cues to derive or guide the corresponding robot actions. This decomposition provides flexibility, but it also makes action prediction dependent on the quality of the generated visual subgoals. Errors in the predicted subgoals may propagate to the inverse dynamics model, causing the predicted actions to deviate from the intended task execution.

Video-feature-conditioned action prediction. A second paradigm keeps the two-stage structure but does not require the video model to generate a complete future observation sequence (Hu et al., 2025; Liang et al., 2025a; Ma et al., 2026; Pai et al., 2026). This design is motivated by the computational cost of diffusion-based models, where producing full future frames may require multiple iterative sampling steps. Instead of decoding a complete video rollout, these methods extract intermediate spatiotemporal features from the video prediction model and use them to condition an action prediction module:

$$f_t = \mathcal{H}\big(u_\theta(o_t, l)\big), \qquad a_t \sim q_\phi(\cdot \mid o_t, f_t),$$

where u_θ denotes the neural backbone of the video prediction model, $\mathcal{H}(\cdot)$ denotes a feature extraction operation, and f_t denotes the extracted spatiotemporal representation. The underlying assumption is that the internal representations of a video model already contain useful information about task intent, scene dynamics, and possible future evolution, even if the model does not explicitly decode the final video during policy inference. This can substantially reduce inference cost compared with full video generation, while still transferring predictive visual representations to the action model. The main limitation is that the interface between the video model and the policy becomes a latent feature rather than an explicit visual plan. As a result, the intermediate representation is less interpretable, harder to inspect, and may not expose whether the policy is using meaningful future information or merely exploiting task-correlated features.

Joint video-action modeling. A third paradigm attempts to model future observations and robot actions jointly within a single generative model (Cheang et al., 2024; Kim et al., 2026; Li et al., 2026; Ye et al., 2026):

$$(o_{t+1:t+H}, a_{t:t+H-1}) \sim p_\theta(\cdot \mid o_t, l).$$

In this formulation, video prediction and action prediction are not treated as two separate modules. Instead, the model learns a joint distribution over visual futures and the corresponding action sequences. A common strategy is to start from a video generation backbone pretrained on large-scale visual data (NVIDIA et al., 2026; Wan Team, 2025), modify the architecture or output space so that it can also produce robot actions, and then adapt the model on robot trajectories with action labels. This unified formulation has the advantage that future observations and actions are learned in a shared representational space, which can improve their consistency: the generated actions are trained together with the

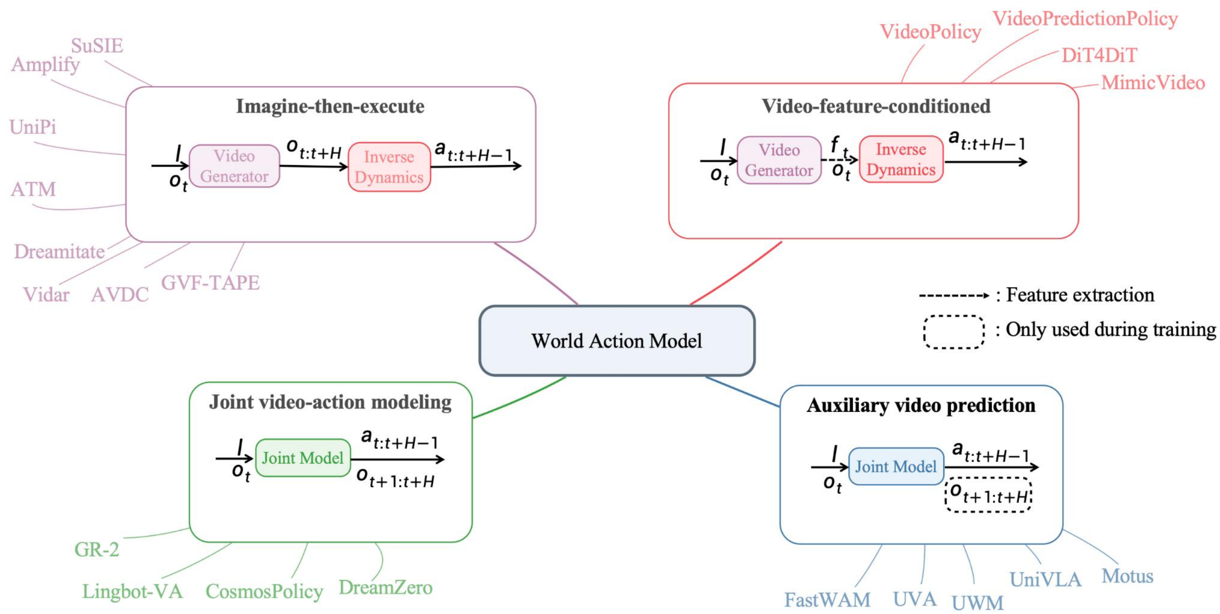

Figure 8. Taxonomy of world action models. Given the observation o_t and language instruction l, world action models couple future observation prediction with robot action generation in different ways. Representative paradigms include imagine-then-execute, video-feature-conditioned action prediction, joint video-action modeling, and auxiliary video prediction for policy learning.

visual future they are supposed to induce. However, joint modeling also inherits the difficulties of both video generation and action learning. It requires robot data with action labels for adapting the action output, and such data are much scarcer than ordinary videos. In addition, the model must jointly handle high-dimensional visual generation and precise action prediction, whose optimization objectives may not always be aligned.

Auxiliary video prediction for policy learning. A fourth paradigm uses video prediction as an auxiliary training objective rather than as an explicit inference-time module (Bi et al., 2026; Bu et al., 2025; Li et al., 2025a; Yuan et al., 2026; Zhu et al., 2025). In this setting, the policy is trained with an additional prediction branch that reconstructs or generates future observations from the policy's internal representation. The purpose of this branch is not to produce an explicit visual plan for execution, but to encourage the policy backbone to encode spatiotemporal information related to future scene evolution. During training, the video prediction loss provides an auxiliary supervisory signal in addition to the action prediction loss. During inference, the auxiliary video branch can be removed, and only the action prediction head is used. This design avoids the cost of generating videos at test time while still using future prediction to shape the learned representation. Its limitation is that the predicted future is not explicitly used as a plan during execution. Therefore, the benefit of the video objective depends on whether the auxiliary prediction task induces representations that are useful for action selection. Balancing the action loss and the video prediction loss can also be nontrivial, since accurate visual prediction does not necessarily lead to better control.

Overall, these four paradigms represent different ways of coupling world prediction with action generation. Imagine-then-execute methods provide modularity and interpretability, but may suffer from a visual-action grounding gap. Feature-conditioned methods reduce the computational cost of explicit video generation, but sacrifice the transparency of visual plans. Joint video-action models offer a unified formulation, but require action-labeled robot data and must jointly optimize visual and action prediction. Auxiliary prediction methods improve efficiency at inference time, but rely on the training objective to implicitly transfer predictive information into the policy. The central design trade-off is therefore how explicitly the future should be represented and how tightly this future representation should be coupled with action prediction.

References

Google DeepMind Blog. URL https://deepmind.google/blog/genie-3-a-new-frontier-for-world-models/. Accessed: 2026-06-05.

Assran, M., Bardes, A., Fan, D., Garrido, Q., Howes, R., Komeili, M., Muckley, M., Rizvi, A., Roberts, C., Sinha, K., Zholus, A., Arnaud, S., Gejji, A., Martin, A., Hogan, F. R., Dugas, D., Bojanowski, P., Khalidov, V., Labatut, P., Massa, F., Szafraniec, M., Krishnakumar, K., Li, Y., Ma, X., Chandar, S., Meier, F., LeCun, Y., Rabbat, M., and Ballas, N. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Bi, H., Tan, H., Xie, S., Wang, Z., Huang, S., Liu, H., Zhao, R., Feng, Y., Xiang, C., Rong, Y., Zhao, H., Liu, H., Su, Z., Ma, L., Su, H., and Zhu, J. Motus: A unified latent action world model. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Black, K., Nakamoto, M., Atreya, P., Walke, H., Finn, C., Kumar, A., and Levine, S. Zero-shot robotic manipulation with pretrained image-editing diffusion models, 2023. URL https://arxiv.org/abs/2310.10639.

Bu, Q., Zeng, J., Chen, L., Yang, Y., Zhou, G., Yan, J., Luo, P., Cui, H., Ma, Y., and Li, H. Closed-loop visuomotor control with generative expectation for robotic manipulation, 2024. URL https://arxiv.org/abs/2409.09016.

Bu, Q., Yang, Y., Cai, J., Gao, S., Ren, G., Yao, M., Luo, P., and Li, H. Univla: Learning to act anywhere with task-centric latent actions. In *Robotics: Science and Systems (RSS)*, 2025. Also at ICLR 2026.

Cheang, C.-L., Chen, G., Jing, Y., Kong, T., Li, H., Li, Y., Liu, Y., Wu, H., Xu, J., Yang, Y., Zhang, H., and Zhu, M. Gr-2: A generative video-language-action model with web-scale knowledge for robot manipulation. *arXiv preprint arXiv:2410.06158*, 2024.

Du, Y., Yang, M., Dai, B., Dai, H., Nachum, O., Tenenbaum, J. B., Schuurmans, D., and Abbeel, P. Learning universal policies via text-guided video generation, 2023. URL https://arxiv.org/abs/2302.00111.

Feng, Y., Tan, H., Mao, X., Xiang, C., Liu, G., Huang, S., Su, H., and Zhu, J. Vidar: Embodied video diffusion model for generalist manipulation. *arXiv preprint arXiv:2507.12898*, 2025.

Gao, S., Zhou, S., Du, Y., Zhang, J., and Gan, C. Adaworld: Learning adaptable world models with latent actions, 2025. URL https://arxiv.org/abs/2503.18938.

Gao, S., Liang, W., Zheng, K., Malik, A., Ye, S., Yu, S., Tseng, W.-C., Dong, Y., Mo, K., Lin, C.-H., Ma, Q., Nah, S., Magne, L., Xiang, J., Xie, Y., Zheng, R., Niu, D., Tan, Y. L., Zentner, K. R., Kurian, G., Indupuru, S., Jannaty, P., Gu, J., Zhang, J., Malik, J., Abbeel, P., Liu, M.-Y., Zhu, Y., Jang, J., and Fan, L. J. Dreamdojo: A generalist robot world model from large-scale human videos, 2026. URL https://arxiv.org/abs/2602.06949.

Goswami, R. G., Krishnamurthy, P., LeCun, Y., and Khorrami, F. Osvi-wm: One-shot visual imitation for unseen tasks using world-model-guided trajectory generation. *arXiv preprint arXiv:2505.20425*, 2025.

Guo, J., Ma, X., Wang, Y., Yang, M., Liu, H., and Li, Q. Flowdreamer: A rgb-d world model with flow-based motion representations for robot manipulation, 2025. URL https://arxiv.org/abs/2505.10075.

Guo, Y., Shi, L. X., Chen, J., and Finn, C. Ctrl-world: A controllable generative world model for robot manipulation, 2026. URL https://arxiv.org/abs/2510.10125.

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

Hansen, N., Su, H., and Wang, X. Td-mpc2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations (ICLR)*, 2024.

Hu, Y., Guo, Y., Wang, P., Chen, X., Wang, Y.-J., Zhang, J., Sreenath, K., Lu, C., and Chen, J. Video prediction policy: A generalist robot policy with predictive visual representations. In *International Conference on Machine Learning (ICML)*, 2025. Spotlight.

Huang, S., Chen, L., Zhou, P., Chen, S., Jiang, Z., Hu, Y., Liao, Y., Gao, P., Li, H., Yao, M., and Ren, G. Enerverse: Envisioning embodied future space for robotics manipulation, 2025a. URL https://arxiv.org/abs/2501.01895.

Huang, S., Chen, Q., Zhang, X., Sun, J., and Schwager, M. Particleformer: A 3d point cloud world model for multi-object, multi-material robotic manipulation, 2025b. URL https://arxiv.org/abs/2506.23126.

Huang, W., Chao, Y.-W., Mousavian, A., Liu, M.-Y., Fox, D., Mo, K., and Fei-Fei, L. Pointworld: Scaling 3d world models for in-the-wild robotic manipulation, 2026. URL https://arxiv.org/abs/2601.03782.

Huang, Y., Zhang, J., Zou, S., Liu, X., Hu, R., and Xu, K. Ladi-wm: A latent diffusion-based world model for predictive manipulation. *arXiv preprint arXiv:2505.11528*, 2025c.

Jeong, Y., Chun, J., Cha, S., and Kim, T. Object-centric world model for language-guided manipulation. *arXiv preprint arXiv:2503.06170*, 2025.

Jiang, H., Hsu, H.-Y., Zhang, K., Yu, H.-N., Wang, S., and Li, Y. Phystwin: Physics-informed reconstruction and simulation of deformable objects from videos. *arXiv preprint arXiv:2503.17973*, 2025.

Kim, M. J., Gao, Y., Lin, T.-Y., Lin, Y.-C., Ge, Y., Lam, G., Liang, P., Song, S., Liu, M.-Y., Finn, C., and Gu, J. Cosmos policy: Fine-tuning video models for visuomotor control and planning. *arXiv preprint arXiv:2601.16163*, 2026. ICLR 2026.

Ko, P.-C., Mao, J., Du, Y., Sun, S.-H., and Tenenbaum, J. B. Learning to act from actionless videos through dense correspondences, 2023. URL https://arxiv.org/abs/2310.08576.

Li, L., Zhang, Q., Luo, Y., Yang, S., Wang, R., Han, F., Yu, M., Gao, Z., Xue, N., Zhu, X., Shen, Y., and Xu, Y. Lingbot-va: Causal world modeling for robot control. *arXiv preprint arXiv:2601.21998*, 2026.

Li, S., Gao, Y., Sadigh, D., and Song, S. Unified video action model. In *Robotics: Science and Systems (RSS)*, 2025a.

Li, W., Zhao, H., Yu, Z., Du, Y., Zou, Q., Hu, R., and Xu, K. Pin-wm: Learning physics-informed world models for non-prehensile manipulation. *arXiv preprint arXiv:2504.16693*, 2025b.

Liang, J., Liu, R., Ozguroglu, E., Sudhakar, S., Dave, A., Tokmakov, P., Song, S., and Vondrick, C. Dreamitate: Real-world visuomotor policy learning via video generation. *arXiv preprint arXiv:2406.16862*, 2024a.

Liang, J., Tokmakov, P., Liu, R., Sudhakar, S., Shah, P., Ambrus, R., and Vondrick, C. Video generators are robot policies, 2025a. URL https://arxiv.org/abs/2508.00795.

Liang, Y., Kumar, N., Tang, H., Weller, A., Tenenbaum, J. B., Silver, T., Henriques, J. F., and Ellis, K. Visualpredicator: Learning abstract world models with neurosymbolic predicates for robot planning. *arXiv preprint arXiv:2410.23156*, 2024b.

Liang, Y., Nguyen, D., Yang, C., Li, T., Tenenbaum, J. B., Rasmussen, C. E., Weller, A., Tavares, Z., Silver, T., and Ellis, K. Exopredicator: Learning abstract models of dynamic worlds for robot planning. *arXiv preprint arXiv:2509.26255*, 2025b. Accepted to ICLR 2026.

Luo, Y. and Du, Y. Grounding video models to actions through goal conditioned exploration, 2025. URL https://arxiv.org/abs/2411.07223.

Ma, T., Zheng, J., Wang, Z., Jiang, C., Cui, A., Liang, J., and Yang, S. Dit4dit: Jointly modeling video dynamics and actions for generalizable robot control. *arXiv preprint arXiv:2603.10448*, 2026.

Maes, L., Le Lidec, Q., Scieur, D., LeCun, Y., and Balestriero, R. Leworldmodel: Stable end-to-end joint-embedding predictive architecture from pixels. *arXiv preprint arXiv:2603.19312*, 2025.

NVIDIA, :, Ali, A., Bai, J., Bala, M., Balaji, Y., Blakeman, A., Cai, T., Cao, J., Cao, T., Cha, E., Chao, Y.-W., Chattopadhyay, P., Chen, M., Chen, Y., Chen, Y., Cheng, S., Cui, Y., Diamond, J., Ding, Y., Fan, J., Fan, L., Feng, L., Ferroni, F., Fidler, S., Fu, X., Gao, R., Ge, Y., Gu, J., Gupta, A., Gururani, S., Hanafi, I. E., Hassani, A., Hao, Z., Huffman, J., Jang, J., Jannaty, P., Kautz, J., Lam, G., Li, X., Li, Z., Liao, M., Lin, C.-H., Lin, T.-Y., Lin, Y.-C., Ling, H., Liu, M.-Y., Liu, X., Lu, Y., Luo, A., Ma, Q., Mao, H., Mo, K., Nah, S., Narang, Y., Panaskar, A., Pavao, L., Pham, T., Ramezanali, M., Reda, F., Reed, S., Ren, X., Shao, H., Shen, Y., Shi, S., Song, S., Stefaniak, B., Sun, S., Tang, S., Tasmeen, S., Tchapmi, L., Tseng, W.-C., Varghese, J., Wang, A. Z., Wang, H., Wang, H., Wang, H., Wang, T.-C., Wei, F., Xu, J., Yang, D., Yang, X., Ye, H., Ye, S., Zeng, X., Zhang, J., Zhang, Q., Zheng, K., Zhu, A., and Zhu, Y. World simulation with video foundation models for physical ai, 2026. URL https://arxiv.org/abs/2511.00062.

Pai, J., Achenbach, L., Montesinos, V., Forrai, B., Mees, O., and Nava, E. mimic-video: Video-action models for generalizable robot control beyond vlas. *arXiv preprint arXiv:2512.15692*, 2026.

Patel, S., Mohan, S., Mai, H., Jain, U., Lazebnik, S., and Li, Y. Robotic manipulation by imitating generated videos without physical demonstrations, 2026. URL https://arxiv.org/abs/2507.00990.

Qi, H., Yin, H., Zhu, A., Du, Y., and Yang, H. Inference-time enhancement of generative robot policies via predictive world modeling, 2026. URL https://arxiv.org/abs/2502.00622.

Shang, Y., Zhang, X., Tang, Y., Jin, L., Gao, C., Wu, W., and Li, Y. Roboscape: Physics-informed embodied world model, 2025. URL https://arxiv.org/abs/2506.23135.

Team, R., Gao, Z., Wang, Q., Zeng, Y., Zhu, J., Cheng, K. L., Li, Y., Wang, H., Xu, Y., Ma, S., Chen, Y., Liu, J., Cheng, Y., Yao, Y., Zhu, J., Meng, Y., Zheng, K., Bai, Q., Chen, J., Shen, Z., Yu, Y., Zhu, X., Shen, Y., and Ouyang, H. Advancing open-source world models, 2026. URL https://arxiv.org/abs/2601.20540.

Wan Team. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Wang, B., Meng, X., Wang, X., Zhu, Z., Ye, A., Wang, Y., Yang, Z., Ni, C., Huang, G., and Wang, X. Embodiedreamer: Advancing real2sim2real transfer for policy training via embodied world modeling. *arXiv preprint arXiv:2507.05198*, 2025a.

Wang, M., Jin, W., Cao, K., Xie, L., and Hong, Y. Contactgaussian-wm: Learning physics-grounded world model from videos. *arXiv preprint arXiv:2602.11021*, 2026.

Wang, S., You, J., Hu, Y., Li, J., and Gao, Y. Skil: Semantic keypoint imitation learning for generalizable data-efficient manipulation. *arXiv preprint arXiv:2501.14400*, 2025b.

Wen, C., Lin, X., So, J., Chen, K., Dou, Q., Gao, Y., and Abbeel, P. Any-point trajectory modeling for policy learning. *arXiv preprint arXiv:2401.00025*, 2024.

World Labs Team. RTFM: A real-time frame model. World Labs Blog, October 2025. URL https://www.worldlabs.ai/blog/rtfm. Accessed: 2026-06-05.

Wu, P., Escontrela, A., Hafner, D., Goldberg, K., and Abbeel, P. Daydreamer: World models for physical robot learning. In *Conference on Robot Learning (CoRL)*, 2022.

Xu, M., Xu, Z., Xu, Y., Chi, C., Wetzstein, G., Veloso, M., and Song, S. Flow as the cross-domain manipulation interface. In *Conference on Robot Learning (CoRL)*, 2024.

Ye, S., Ge, Y., Zheng, K., Gao, S., Yu, S., Kurian, G., Indupuru, S., Tan, Y. L., Zhu, C., Xiang, J., Malik, A., Lee, K., Liang, W., Ranawaka, N., Gu, J., Xu, Y., Wang, G., Hu, F., Narayan, A., Bjorck, J., Wang, J., Kim, G., Niu, D., Zheng, R., Xie, Y., Wu, J., Wang, Q., Julian, R., Xu, D., Du, Y., Chebotar, Y., Reed, S., Kautz, J., Zhu, Y., Fan, L., and Jang, J. Dreamzero: World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026.

Yin, T., Mei, Z., Sun, T., Zha, L., Zhou, E., Bao, J., Yamane, M., Shorinwa, O., and Majumdar, A. Womap: World models for embodied open-vocabulary object localization. *arXiv preprint arXiv:2506.01600*, 2025.

Yuan, T., Dong, Z., Liu, Y., and Zhao, H. Fast-wam: Do world action models need test-time future imagination? *arXiv preprint arXiv:2603.16666*, 2026.

Zhang, C., Zhang, X., Pan, W., Zheng, L., and Zhang, W. Generative visual foresight meets task-agnostic pose estimation in robotic table-top manipulation, 2025. URL https://arxiv.org/abs/2509.00361.

Zhen, H., Sun, Q., Zhang, H., Li, J., Zhou, S., Du, Y., and Gan, C. Tesseract: Learning 4d embodied world models, 2025. URL https://arxiv.org/abs/2504.20995.

Zhi, H., Chen, P., Zhou, S., Dong, Y., Wu, Q., Han, L., and Tan, M. 3dflowaction: Learning cross-embodiment manipulation from 3d flow world model. *arXiv preprint arXiv:2506.06199*, 2025.

Zhou, G., Pan, H., LeCun, Y., and Pinto, L. Dino-wm: World models on pre-trained visual features enable zero-shot planning. *arXiv preprint arXiv:2411.04983*, 2024.

Zhu, C., Yu, R., Feng, S., Burchfiel, B., Shah, P., and Gupta, A. Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets. In *Robotics: Science and Systems (RSS)*, 2025.