World Models and World-Action Models: An Accessible and Comprehensive Survey

Inkyu Sa, Chef Robotics
inkyu@chefrobotics.ai

Abstract*—A *world model* is a learned internal simulator: given the current situation and a proposed action, it predicts what happens next, so that an artificial agent can "imagine" the consequences of its choices before committing to them. This single idea (an internal model of reality that can be run forward in the mind) now unifies a remarkably broad range of modern artificial intelligence, from game-playing agents that master Atari from two hours of experience, to robots that learn manipulation by dreaming, to systems that generate playable video games and drivable streets frame by frame. This survey provides an *accessible yet technically complete* review of world models and the emerging class of *world-action models* that both imagine the future and output the actions to reach it. Covering more than 220 research contributions (including peer-reviewed papers, influential technical reports, and industry blog posts through mid-2026), we organize the field along a primary axis of *modeling approach* (latent-recurrent state-space models, value-equivalent planning models, autoregressive-token transformers, diffusion and generative-frame models, joint-embedding predictive architectures, and large video-generation foundation models) and a secondary axis of *application domain* (reinforcement learning and control, robotics, autonomous driving, games and interactive environments, general-purpose simulation, and language, code, and agentic environments). We present a unifying taxonomy, six figures, and ten comparison tables covering more than 135 systems together with more than 20 benchmarks and metrics, together with a critical treatment of evaluation, including the sharp debate over whether large video generators truly "understand" physics. Every concept is introduced with plain-language intuition and analogy before its formal definition, making the survey suitable for readers with only a first-year-undergraduate technical background while remaining rigorous for specialists. We close with seven concrete research directions. The central thread throughout is a single question that distinguishes every method: *what should a world model predict, and should it also generate robot actions?

***Index Terms*—World models, world-action models, model-based reinforcement learning, video generation, diffusion models, embodied AI, robot learning**

I. Introduction

In 1943, long before the first digital computer ran a program, the psychologist Kenneth Craik proposed that the human mind works by carrying "a small-scale model of external reality" inside the head, a model it can run to try out alternatives, anticipate what will happen, and choose the best course of action before acting [1]. When you picture how a chess move will play out before touching the piece, or rehearse a difficult conversation in your head, you are running exactly such a model. This is the intuition at the heart of the world model: an internal simulator of how the environment evolves, learned from experience and used to predict, plan, and imagine.

Intuition: A world model is an agent's imagination. Instead of only reacting to what it senses right now, an agent with a world model can ask "what would happen if I did this?" and answer the question internally (safely, cheaply, and quickly) before acting in the real world.

In artificial intelligence, this idea first became concrete in reinforcement learning (RL), the study of agents that learn by trial and error. The field draws a sharp line between model-free agents, which learn purely by acting and observing rewards without ever representing how the world works, and model-based agents, which first learn a model of the environment's dynamics and then use it to plan [8]. The promise of the model-based route is enormous sample efficiency: an agent that has learned how the world responds to its actions can practice thousands of times in its imagination for every real attempt. Richard Sutton's Dyna architecture [9] made this explicit around 1990, interleaving real experience, model learning, and "planning" as practice on simulated experience. In the same period, Jürgen Schmidhuber formulated a neural version: a recurrent network that predicts the future, queried by a separate controller that improves its plan by tracing how each imagined outcome depends on the actions that produced it (differentiating through the imagined rollout) [10], [11].

The modern era of learned world models is usually dated to 2018, when Ha and Schmidhuber's paper simply titled *World Models* [12] vividly demonstrated that an agent could learn a competitive policy while training almost entirely inside its own dream, a learned simulator of a racing game and a shooter. The years since have seen an explosion. The Dreamer lineage [4], [13]–[15] turned latent imagination into a reliable engineering recipe, culminating in DreamerV3 [4], published in *Nature* in 2025, which mined a diamond in Minecraft from scratch with no human demonstrations. MuZero [16] mastered Go, chess, shogi, and Atari with a model that predicts only what matters for decisions and cannot even draw the screen. By 2024 to 2026, world models had grown into internet-scale systems, several of which Fig. 1 shows side by side: OpenAI's Sora [6] argued that scaled video generators are "world simulators"; Google DeepMind's Genie [5], [17], [18] generated playable worlds from a single image or text prompt; NVIDIA's Cosmos [19] shipped open "world foundation models" for physical AI; Meta's V-JEPA 2 [20] enabled zero-shot robot

Preprint, version of 12 August 2026. This manuscript has not been peer reviewed. The author is affiliated with Chef Robotics; the views expressed are the author's own, and no system described in this survey is endorsed on the basis of that affiliation.

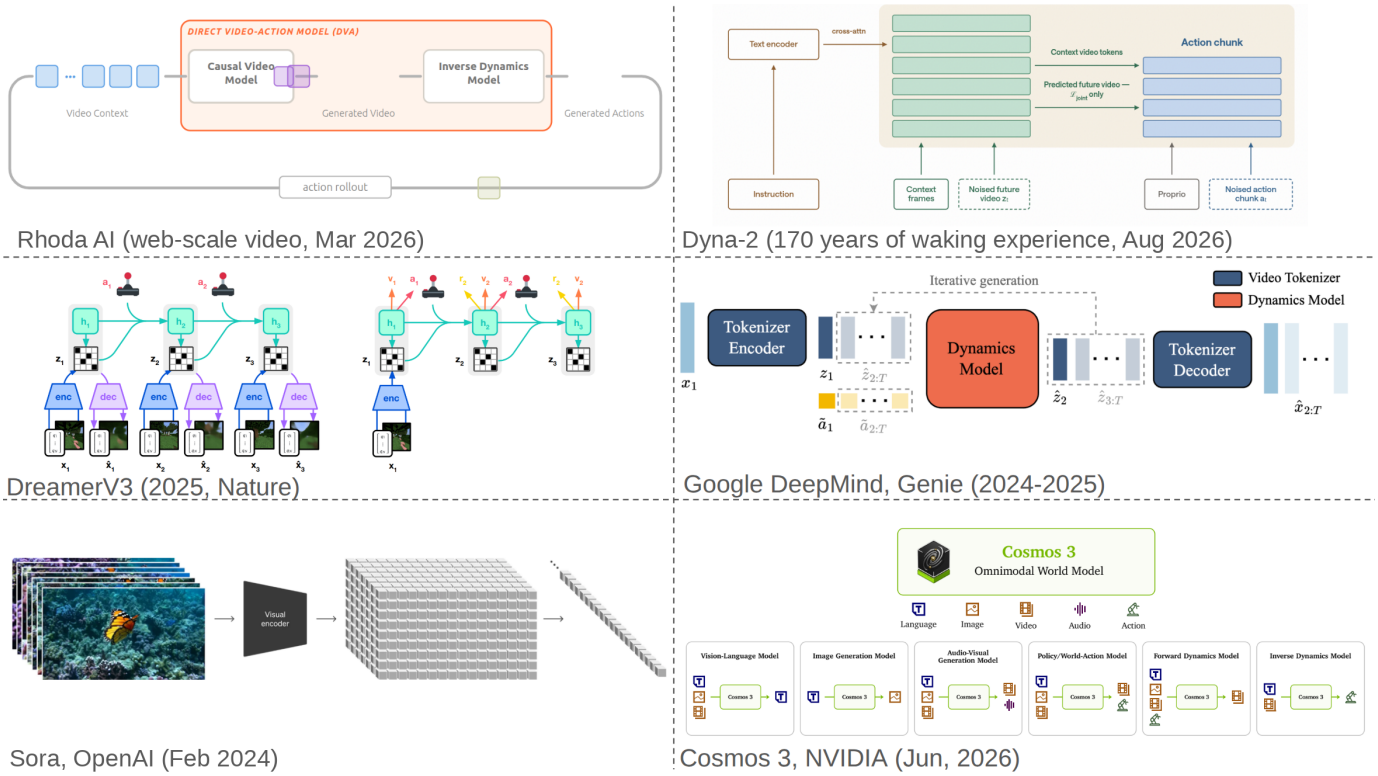

Fig. 1: Six representative world models, one panel each, spanning the range this survey covers. *Top left:* Rhoda's Direct Video-Action model [2], in which a causal video model's prediction is converted to actions by an inverse-dynamics model and fed back as an action rollout (Section V-B). *Top right:* Dyna-2 [3], whose video and action streams are tokenized separately and cross-attend, with instruction and proprioception as side inputs (Section V-B). *Middle left:* DreamerV3's recurrent state-space model [4], encoding each frame to a categorical latent, rolling it forward through $\mathbf{h}_t$, and decoding back to pixels (Section IV-A). *Middle right:* Genie's tokenizer-plus-dynamics loop [5], generating the next latent iteratively from latent actions (Section IV-C). *Bottom left:* Sora's spacetime-patch encoder [6], which flattens video into a sequence of latent patches (Section IV-F). *Bottom right:* Cosmos 3 [7], one omnimodal model over language, image, video, audio, and action that is configured into several roles, including a world-action policy (Section IV-F). Panels are grouped by family rather than by date, and together they trace the survey's central progression from passive video generation toward action-conditioned and omnimodal systems. Each panel is adapted from the corresponding cited source, to which all credit for the original diagrams belongs.

planning; and Microsoft's Muse/WHAM [21], a "World and Human Action Model," reached *Nature* for modeling both a game world and the human actions within it.

This momentum has spilled out of the laboratory. By 2026, world models had become a live dividing line in the race to build physical AI: robots general enough to reach a "ChatGPT moment." The field split into camps: one building dedicated world-model labs (at video and robotics companies alike), a second betting instead on vision-language-action (VLA) controllers [22], and a third seeking to combine the two into what this survey calls a world-action model (Section I-A). We return to this industry debate in Section VIII-C. Part of what makes a clear, accessible account timely is that the field's own practitioners disagree about what a world model is and what it is for.

A. World Models versus World-Action Models

This survey foregrounds a distinction that increasingly organizes the field. A world model (in the narrow sense) answers a predictive question: given the current state and an action, what happens next? It is a simulator. A world-action model goes further: it both imagines a desired future and outputs the actions needed to reach it, unifying the simulator and the policy in one system. Microsoft's WHAM [21] generates game frames and controller inputs; Meta's V-JEPA 2-AC [20] plans robot actions inside a learned latent world; NVIDIA's emerging "world-action model" framing [23] pretrains a large video model to imagine, then fine-tunes it to act. Throughout the survey we call out this distinction (whether a system merely predicts or also acts) family by family and in the comparison tables, because it cuts across every architectural family and every application domain. This two-part question (what should a world model predict, and should it also generates actions?) is the central thread of this survey: its first half organizes the six modeling approaches (Section IV), while its second separates world models from world-action models.

B. Why This Survey, and How It Differs

The world-model literature is now served by several surveys and position pieces. Ding *et al.* [24] organize the field around

"understanding versus predicting"; Zhu *et al.* [25] focus on the "is Sora a world simulator?" question; domain-specific surveys cover autonomous driving [26] and robot learning [27]; and position essays stake out the field's central architectural debate, most prominently LeCun's manifesto for latent-predictive world models [28] against the pixel-generative view, alongside more recent critiques [29] and a functional taxonomy of world models [30]. A 2026 perspective proposes a scientific definition and a staged roadmap [31]; its framing of world models as internal simulators of environment structure and dynamics converges with our own. These works are valuable but share three limitations for a newcomer: they are written for expert machine-learning audiences, they are typically organized around a single abstract dichotomy, and most predate the wave of real-time interactive systems (Genie 3, Cosmos, Runway GWM-1, WHAMM) that arrived in 2025 and 2026 and has reshaped the field.

This survey differs in four ways. First, it is accessible: every technical term is introduced with a plain-language gloss and, where helpful, an everyday analogy, before the formal definition and equations, so a curious first-year undergraduate can follow the argument without sacrificing the rigor a specialist expects. Second, it is unifying: it places latent-state RL models, value-equivalent planners, token transformers, diffusion engines, joint-embedding architectures, and video foundation models under one taxonomy (Fig. 2), then shows how the same six approaches serve six different application domains. Third, it foregrounds the world model versus world-action model distinction as a first-class axis. Fourth, it is current, covering developments through mid-2026 and treating the "video generators as world models" debate critically, weighing the emergent-simulator claim against rigorous physics-benchmark rebuttals [32], [33].

The main contributions of this survey are:

- A unifying taxonomy of world models and world-action models organized by modeling approach (six families) and application domain (six domains), presented as a visual table of contents in Fig. 2 and used consistently throughout.
- Ten comparison tables covering more than 135 systems, plus more than 20 benchmarks and metrics, across architecture, prediction target, action-conditioning, domain, headline results, and venue/year.
- A critical review of evaluation, distinguishing sample-efficiency benchmarks, generative-video metrics, and the new physics- and controllability-aware benchmarks, and explaining why visual realism and genuine world understanding are only weakly correlated.
- Seven concrete research directions, ordered by importance and grounded in the field's open problems: physical and causal consistency, long-horizon coherence, the prediction-target question, unified world-action models, honest evaluation, real-time inference, and safety.

The remainder of this paper is organized as follows. Section II defines world models formally, introduces our notation, presents the taxonomy, and states the survey's scope. Section III reviews the prerequisite concepts (reinforcement learning, generative modeling, and self-supervised learning) and traces the historical timeline. Section IV is the core of the survey, covering the six modeling families in turn, each with a comparison table. Section V organizes the same landscape by application domain. Section VI synthesizes how a world model becomes a world-action model. Section VII surveys benchmarks and evaluation protocols. Section VIII concludes with the state of the art, key findings, the world-model-versus-VLA rift in physical AI, open challenges, and research directions. Section IX provides a glossary of abbreviations.

TABLE I: Summary of notation used in this survey.

Symbol	Description
$\mathbf{s}_t$	Environment state at time t
$\mathbf{o}_t$	Observation (*e.g.*image) at time t
$\mathbf{a}_t$	Action taken at time t
r_t	Reward received at time t
$\mathbf{z}_t$	Latent (compressed) state at time t
$\mathbf{h}_t$	Deterministic recurrent state
γ	Discount factor, $\gamma \in [0, 1)$
π_θ	Policy (action selector) with parameters θ
p_ϕ	World model with parameters ϕ
V, Q	State-value and action-value functions
T	Transition (dynamics) function
H	Planning / imagination horizon
$\mathbf{g}$	Goal (*e.g.*a goal image or instruction)
$\mathcal{D}$	Dataset of experience / demonstrations
e_ψ	Encoder mapping observations to latents

II. Problem Definition, Notation, and Taxonomy

We begin by making the intuitive notion of a world model precise. The notation introduced here is used consistently throughout the survey; Table I summarizes it.

A. The Environment: Markov Decision Processes

Most world-model research assumes the environment is a Markov Decision Process (MDP), the standard mathematical model of sequential decision-making.

Intuition: An MDP is just a formal way of saying: "the world is in some state; you take an action; the world moves to a new state and hands you a reward; repeat." The word Markov means the next state depends only on the current state and action, not the entire history, like a board game where the current board tells you everything you need to know.

Formally, an MDP is a tuple $(\mathcal{S}, \mathcal{A}, T, R, \gamma)$: a set of states $\mathcal{S}$, a set of actions $\mathcal{A}$, a transition function $T(\mathbf{s}_{t+1} \mid \mathbf{s}_t, \mathbf{a}_t)$ giving the probability of the next state, a reward function $R(\mathbf{s}_t, \mathbf{a}_t)$, and a discount factor γ that makes near-term reward worth more than distant reward. The agent's goal is to learn a policy $\pi_\theta(\mathbf{a}_t \mid \mathbf{s}_t)$ that maximizes the expected discounted return $\mathbb{E}[\sum_t \gamma^t r_t]$.

In most interesting settings the agent does not see the true state $\mathbf{s}_t$ directly but only a high-dimensional observation $\mathbf{o}_t$, such as a camera image, giving a partially observable MDP. This is why many world models first compress observations into a compact latent state before modeling dynamics.

B. What Is a World Model?

Formally, a world model is a learned approximation p_ϕ of the environment's dynamics. In its most general action-conditioned form it predicts the next state (or observation), and often the reward, given the current state and action:

$$p_\phi(\mathbf{s}_{t+1}, r_t \mid \mathbf{s}_t, \mathbf{a}_t) \;\approx\; T(\mathbf{s}_{t+1} \mid \mathbf{s}_t, \mathbf{a}_t)\, p(r_t \mid \mathbf{s}_t, \mathbf{a}_t, \mathbf{s}_{t+1}). \tag{1}$$

By composing this one-step model, the agent can roll out an imagined trajectory of length H from a starting state under a candidate action sequence, without touching the real environment. The critical design choices, which distinguish every method in this survey, are: (i) what space the prediction lives in (raw pixels, a compressed latent, an abstract representation, or just reward and value); (ii) how the model is trained (reconstruction, next-token prediction, denoising, or feature prediction); and (iii) whether the model is action-conditioned and interactive at all.

A passive predictor that forecasts future frames without an action input (much of video generation) is a world model only in a weak sense. An action-conditioned model that answers "what happens if I do $\mathbf{a}_t$?" is a world model in the full sense. And a world-action model additionally produces the policy π_θ itself, so that one system both imagines and acts.

Beyond these engineering motivations, there is a theoretical argument that world models are unavoidable. Richens *et al.* [34] prove, within a formal model of goal-directed competence, that any agent able to generalize across multi-step, goal-directed tasks must have implicitly learned a predictive model of its environment, that this model can in principle be extracted from the agent's policy, and that agents which reach harder goals must carry more accurate models. On this view a world model is not one design option among many: it is what competent goal-directed behavior necessarily contains, whether or not the designer put it there deliberately.

That result came with two strong assumptions, however: the agent is deterministic and the environment fully observable. Neither holds for the systems in this survey. Cifuentes [35] removes both, extending the theorem to stochastic agents in partially observable environments and showing that a stochastic agent cannot escape learning its environment through the very randomization it uses to act; they also weaken the generality requirement, so that agents less capable than Richens *et al.* demanded are still obliged to carry a model. The necessity claim is therefore stronger and more general than when it was first made.

Intuition: The claim is not that engineers must build a world model, but that a capable agent ends up containing one whether or not anybody intended it. If a system reliably achieves varied multi-step goals, something inside it is already predicting consequences.

Honesty requires the counterweight. A necessity theorem says a competent agent contains a model; it does not say that a model which predicts well has recovered how the world actually works. Vafa *et al.* [36] probe exactly this gap and find it real. Their inductive bias probe fine-tunes a pretrained sequence model on many small tasks derived from a postulated world model and asks whether it extrapolates as though it had recovered that model. A 109M-parameter transformer trained on simulated planetary orbits predicts held-out trajectories almost perfectly ($R^2 > 0.9999$), yet probing shows its internal bias is not toward Newtonian state: it has learned to reproduce the observations without the mechanics that generate them. The experiments use purpose-built models on synthetic sequences rather than large video generators, so the result is a proof of possibility rather than a measurement of today's simulators. Held together, the two results bracket the field's central uncertainty. Competence implies a model; accurate prediction does not imply a correct one. Section IV-F shows the same tension play out empirically in video generators.

C. Three Uses of a World Model

Once an agent has a world model, it can be put to three distinct uses. We return to this framing throughout:

1) Planning. At decision time, roll out candidate action sequences in the model, score the imagined outcomes, and execute the best. This is the idea behind model-predictive control (MPC) and tree search: the agent thinks before it acts.
2) Learning in imagination. Train a policy on trajectories generated by the model rather than the real environment, so that the agent needs far less real interaction, which is slow, expensive, and often dangerous. Dreamer is the canonical instance.
3) Data generation. Use the model to synthesize large amounts of realistic training data ("dreamed" rollouts), cheaply expanding a small real dataset. Robotics and driving lean hardest on this route.

D. A Taxonomy of World Models

Figure 2 presents the taxonomy that structures this survey. We organize the field along two axes. The **primary axis** is the modeling approach, the answer a method gives to "what should the model predict, and how?" We identify six families:

- Latent-recurrent / state-space (RSSM) models (Section IV-A) compress observations into a compact latent state and roll it forward with a recurrent network; *e.g.*PlaNet, the Dreamer series.
- Value-equivalent / planning models (Section IV-B) predict only reward, value, and policy (exactly what a planner needs), ignoring appearance; *e.g.*MuZero, EfficientZero.
- Autoregressive-token / transformer models (Section IV-C) treat the world like language: tokenize observations and predict the next token; *e.g.*IRIS, Genie 1.
- Diffusion / generative-frame models (Section IV-D) render the future pixel-by-pixel via iterative denoising, enabling photorealistic neural game engines; *e.g.*GameNGen, DIAMOND, Oasis.
- JEPA / joint-embedding models (Section IV-E) predict in an abstract representation space rather than pixels; *e.g.*V-JEPA 2, DINO-WM.
- Video-generation foundation models (Section IV-F) scale internet-video generation and are framed as general "world simulators"; *e.g.*Sora, Cosmos, Veo.

The **secondary axis** is the application domain: reinforcement learning and control, robotics (including world-action models), autonomous driving, games and interactive environments, general-purpose simulation and spatial intelligence, and language, code, and agentic environments (Section V). Crucially, the two axes are largely independent: almost any modeling approach can serve almost any domain, a central message of this survey. One scope note: the primary axis classifies numeric prediction targets, whether pixels, tokens, latents, or value. Symbolic world models, which predict a program or a natural-language state description, sit outside it and we treat them as a domain instead (Section V-F); Section V-F returns to why that target resists the axis.

E. Scope of This Survey

This survey covers learned, predictive models of environment dynamics and their use for planning, policy learning, data generation, and world generation. We include systems that predict in pixel space, latent space, abstract-representation space, and reward/value space; passive video predictors are included only where they are framed and used as world simulators. We treat model-based RL, robotics, autonomous driving, games, general video/3D simulation, and the symbolic environments of language and code.

We deliberately bound the scope. Classical model-based control with hand-specified dynamics (optimal control, system identification) is touched on only as historical grounding; we point readers to standard texts. Pure VLA policies without a predictive world-model component are outside our scope, though we note where world models augment them. We do not attempt exhaustive coverage of general video generation for entertainment except where it bears on the world-simulator question. Within these bounds, we aim for breadth across approaches and domains and depth on the most influential systems.

III. Background: The Ingredients of a World Model

Every world model combines ideas from three areas: reinforcement learning (which motivates why we want a model and how to use it), generative modeling (which supplies the machinery to predict the future), and self-supervised representation learning (which lets a model decide what to predict). This section reviews each at a level accessible to a general technical reader, then places the field's milestones on a timeline.

A. Reinforcement Learning in One Page

In reinforcement learning, an agent repeatedly observes the environment, takes an action, and receives a reward, seeking to maximize its total reward over time (Section II-A). Two quantities are central. The state-value $V^\pi(\mathbf{s})$ is the expected total future reward starting from state $\mathbf{s}$ and following policy π; the action-value $Q^\pi(\mathbf{s}, \mathbf{a})$ is the same but for taking action $\mathbf{a}$ first. Both obey the Bellman equation.

Intuition: The value of a situation is "what I get now, plus a slightly discounted version of how good the next situation will be." This recursive bookkeeping is how an agent assigns credit to actions whose payoff comes much later.

Formally, the Bellman equation relates the value of a state to the reward plus the discounted value of the next state:

$$Q^\pi(\mathbf{s}_t, \mathbf{a}_t) = \mathbb{E}[\, r_t + \gamma\, V^\pi(\mathbf{s}_{t+1})\,]\,. \tag{2}$$

Model-free methods estimate V or Q (or a policy directly) purely from sampled experience. Model-based methods instead learn the transition function T and reward R (that is, a world model) and use it to plan or to generate imagined experience. The trade-off is fundamental: model-free methods are simple and unbiased but data-hungry, while model-based methods can be far more sample-efficient but suffer from model bias. Model bias means errors in the learned model that a planner can exploit, producing plans that succeed in the dream but fail in reality. Managing this bias is a recurring theme, whether through uncertainty (PILCO [37], and its deep-network successor PETS [38], which plans by model-predictive control through an ensemble of probabilistic networks so that disagreement among ensemble members flags states the model does not understand), short rollouts (MBPO [39]), or value-oriented training (MuZero [16]).

Intuition: An ensemble is a committee of models trained on the same data. Where they agree, the model is probably right; where they disagree, the agent is extrapolating and should not trust its own dream. This is the simplest useful measure of a world model's own uncertainty, and it recurs throughout the survey. Sutton's Dyna [9] (Section I) is the canonical blueprint unifying the two.

B. The Generative Modeling Toolbox

To predict a future observation, a world model needs a way to represent and generate high-dimensional data such as images. Four families of tools recur.

1) Variational Autoencoders (VAEs): A VAE [40] learns to compress an observation into a small latent vector $\mathbf{z}$ (via an encoder) and reconstruct it (via a decoder), while keeping the latent space smooth. World models use the encoder to shrink pixels into a manageable latent state; generative adversarial networks (GANs) [41] are an alternative image generator, though VAEs, tokenized autoregression, and the diffusion models below dominate modern world models.

Intuition: A VAE is a learned zip-and-unzip for images: it finds a short code that captures the gist of a picture and can regenerate the picture from the code. Working in the short-code space is far cheaper than working in pixels.

2) Autoregressive models and tokenization: Large language models predict text one token (word-piece) at a time, each conditioned on all previous tokens, using the Transformer architecture [42]. The same architecture also processes images once they are split into patches (the Vision Transformer [43]), and video once frames are turned into discrete tokens by a vector-quantized autoencoder (VQ-VAE [44], VQGAN [45]). An image becomes a short grid of "visual words," and a Transformer predicts the next word. MaskGIT [46] accelerates

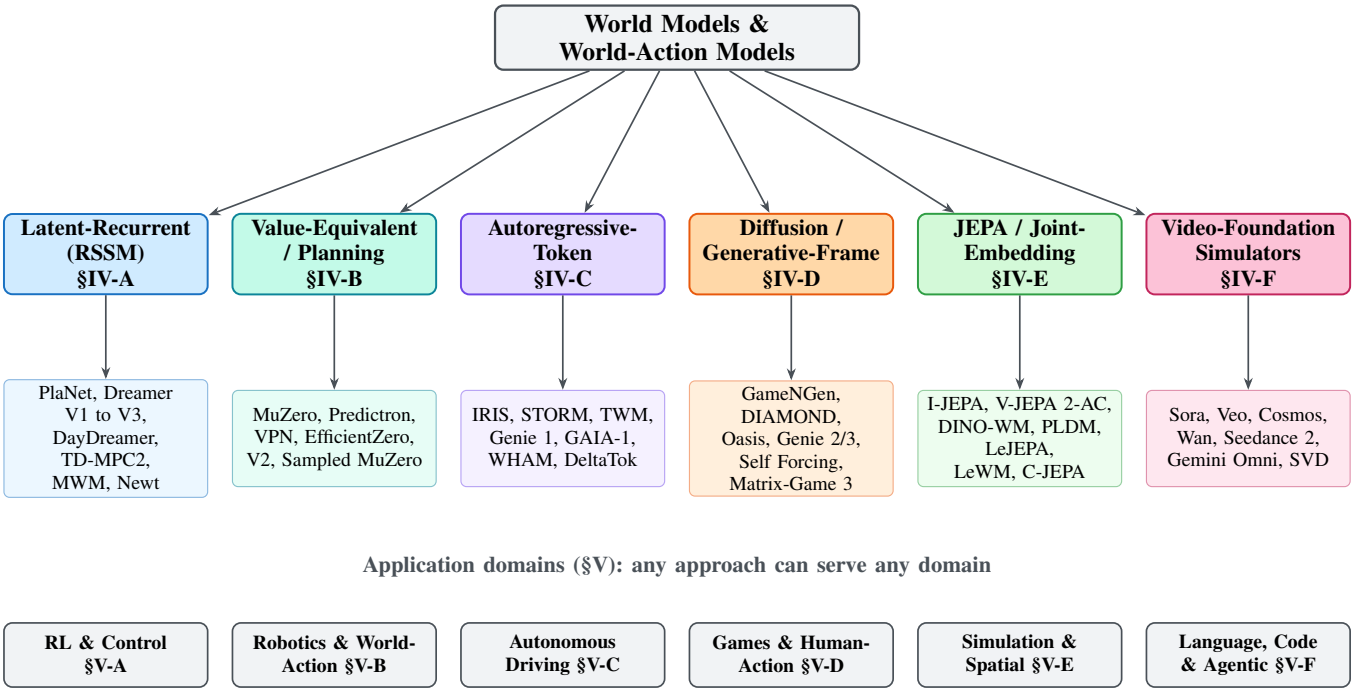

Fig. 2: Taxonomy of world models and world-action models. The *primary axis* (top) is the modeling approach: six families defined by what the model predicts and how it is trained. The *secondary axis* (bottom ribbon) is the application domain. The two axes are largely independent: any modeling approach can, in principle, serve any domain, which is why the survey treats them separately (Sections IV and V). Representative systems are listed under each family.

this by predicting many tokens in parallel. This underpins the autoregressive-token family (Section IV-C).

Intuition: A Transformer works by attention: to predict the next token, it lets that token "look at" every earlier token and weigh how relevant each one is, rather than reading through a single fixed-size memory. That is why Transformers capture long-range structure so well, whether the tokens are words in a sentence or patches of a video.

3) Diffusion and flow models: Diffusion models [47], [48] generate an image by starting from pure noise and iteratively denoising it into a clean sample; latent diffusion [49] does this in a compressed space for efficiency. Flow matching [50] is a closely related, often faster alternative that learns a smooth "velocity field" transporting noise to data. Diffusion is the leading approach to high-fidelity image and video synthesis, and hence to photorealistic generative-frame world models (Section IV-D).

Intuition: Diffusion is like sculpting from static: begin with a block of random noise and repeatedly refine it, each step removing a little noise, until a coherent scene emerges. Conditioning the refinement on the previous frames and the agent's action turns this into a next-frame predictor.

4) Contrastive and self-supervised learning: Rather than reconstruct pixels, a model can learn representations by predicting relationships in feature space, *e.g.*matching images to text (CLIP [51]) or distinguishing real from corrupted inputs. Frozen self-supervised features such as DINOv2 [52] provide rich, object-aware image descriptions that some world models predict directly instead of pixels (Section IV-E).

C. Self-Supervised World Models and the Prediction-Target Debate

The single deepest disagreement in the field concerns what a world model should predict. Pixel-generative models (diffusion, autoregressive tokens) predict the full future observation, which is visually rich but wastes capacity on unpredictable detail (every leaf, every reflection) and is expensive to sample. Value-equivalent models predict only reward and value, which is maximally efficient for planning but produces no viewable rollout. Joint-embedding predictive architectures (JEPA) [28] stake out a middle ground: predict an abstract representation of the future, discarding unpredictable nuisance detail while keeping the structure needed to act. We treat this debate (pixels versus representations versus value) as a unifying thread, visualize it as a spectrum in Fig. 6, and revisit it directly in Section VIII-B.

D. Historical Foundations and Timeline

The intellectual lineage of world models spans cognitive science, neuroscience, control theory, and deep learning. Craik's mental-model hypothesis [1] (1943) is the conceptual origin. In neuroscience, the free-energy principle [53] casts the brain as a generative world model that acts to minimize prediction error. In control, PILCO [37] (2011) showed that an uncertainty-aware dynamics model yields extreme data efficiency, learning

cart-pole swing-up on a real robot from under 18 seconds of data. In deep learning, Schmidhuber's recurrent controller-model systems [10], [11] and Ha and Schmidhuber's *World Models* [12] established the modern template, and LeCun's 2022 manifesto [28] placed a predictive world model at the center of a blueprint for autonomous machine intelligence. Figure 3 traces the milestones from 2011 to 2026, color-coded by the taxonomy families of Fig. 2. With these reinforcement-learning, generative, and self-supervised ingredients in place, we now examine how they combine into the six modeling families.

IV. Modeling Approaches

This section, the core of the survey, reviews the six modeling families of Fig. 2 in turn. Each family answers our organizing question (what should the model predict, and how?) differently, and each carries characteristic strengths and weaknesses. Figure 4 contrasts the mechanisms side by side; each subsection closes with a comparison table.

A. Latent-Recurrent State-Space Models (RSSM)

This family, the workhorse of model-based RL, learns a compact latent state directly from observations and rolls it forward with a recurrent network, then trains a policy by "imagining" inside that latent space. The defining architecture is the Recurrent State-Space Model (RSSM), which splits the latent state into a deterministic recurrent path and a stochastic path. The deterministic path is a gated recurrent unit (GRU), a kind of recurrent neural network (RNN) that carries information reliably across many steps; the stochastic path is a learned distribution that captures uncertainty and multiple possible futures.

Intuition: An RSSM keeps two kinds of memory: a steady one that reliably remembers facts over time, and a fuzzy one that admits the future is uncertain. Planning happens entirely in this small latent world, never in raw pixels, like planning a road trip on a simple map rather than by imagining every blade of grass along the way.

PlaNet [13] introduced the RSSM and showed that planning purely in a learned latent space (via the cross-entropy method) matches top model-free agents on visual control tasks while using roughly 200× fewer episodes. Dreamer (V1) [14] replaced expensive online planning with a learned actor-critic trained entirely on imagined latent rollouts: because the world model is differentiable, gradients of the value function flow back through the imagined trajectory to improve the policy directly. DreamerV2 [15] adapted the recipe to the jumpy, discrete world of Atari by replacing the Gaussian latent with a vector of categorical latents, becoming the first world-model agent to reach human-level Atari. DreamerV3 [4], published in *Nature*, is the current reference point: a single algorithm with one fixed set of hyperparameters masters 150+ tasks spanning continuous and discrete actions and 2D and 3D worlds, thanks to robustness techniques (symlog transforms, two-hot value heads, and KL balancing) that handle widely varying reward and value magnitudes. Famously, out of the box it became the first program to collect a diamond in Minecraft with no human data or curriculum.

Successors specialize the recipe. DayDreamer [54] applied Dreamer directly to physical robots with no simulator, teaching a quadruped to walk from scratch in about one hour of real time. Director [55] added hierarchy, with a high-level "manager" proposing latent goals for a low-level "worker" to reach, solving sparse-reward tasks that defeat flat agents. Dreaming [56] removed the pixel decoder entirely, using a contrastive objective to avoid the "object vanishing" failure where reconstruction ignores small but crucial objects. MWM [57] attacks the same weakness from the opposite direction, keeping reconstruction but separating it from dynamics: a masked autoencoder with a vision transformer learns representations by reconstructing pixels from masked convolutional features, and a latent dynamics model is trained on top of those representations rather than jointly with them. Decoupling the two raises success on 50 visual Meta-World manipulation tasks from 67.9% to 81.7%, evidence that much of the small-object failure is an optimization conflict between representation and dynamics rather than a flaw in reconstruction itself. SLAC [58] pairs a latent-variable model with off-policy control, and Plan2Explore [59] learns the world model itself in a task-agnostic way, by seeking states where the model is most uncertain. TD-MPC [60] and TD-MPC2 [61] sit at the boundary with the value-equivalent family: they learn a decoder-free, reward-and-value-predictive latent model and combine short-horizon planning with a learned terminal value function; TD-MPC2 scales this to a single 317M-parameter agent performing 80 tasks across multiple robot bodies. Newt [62] carries that line further and is the family's clearest test of whether the foundation-model recipe transfers to model-based control: a language-conditioned, decoder-free latent world model is pretrained on demonstrations and then optimized jointly with online interaction across 200 tasks at once, spanning ten domains from DMControl and Meta-World to Atari. It ships MMBench, a benchmark for massively multitask reinforcement learning assembled from 159 existing tasks plus 41 new ones. Finally, Dreamer 4 [63] marks the family's convergence with modern generative modeling: it replaces the RNN with an efficient Transformer trained by a diffusion-style "shortcut forcing" objective, runs as a real-time interactive simulator at $\sim$21 FPS, and trains an agent entirely offline. It is the first system reported to mine a Minecraft diamond from a fixed dataset with no environment interaction at all, using $\sim$100× less data than a comparable large-scale baseline (OpenAI's VPT [64]).

Table II compares the family. Its enduring advantages are extreme sample efficiency and generality; its historical limitations (a reliance on pixel reconstruction and on recurrent memory) are exactly what the value-equivalent, token, diffusion, and JEPA families set out to address.

B. Value-Equivalent and Planning-Oriented Models

This family rests on a provocative insight: a world model used for planning does not need to reconstruct the world at all; it only needs to be right about the quantities a planner

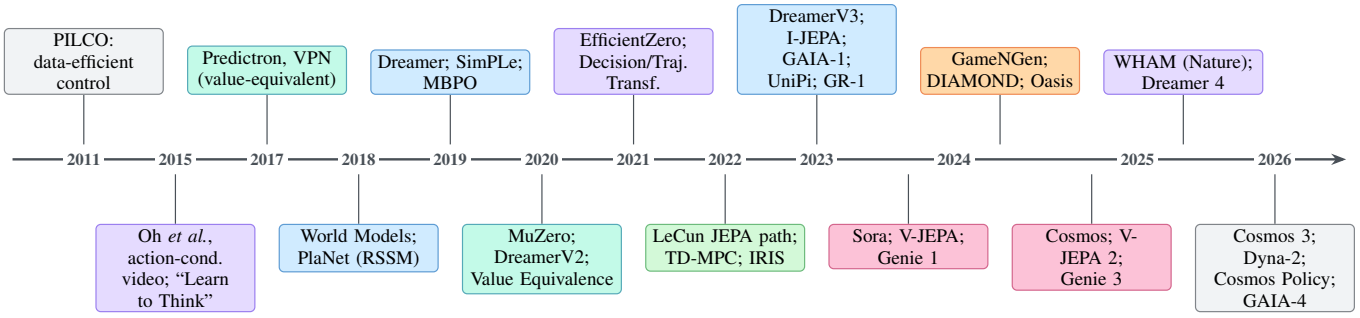

Fig. 3: Timeline of world-model milestones, 2011 to 2026. Each node's color marks the taxonomy family it best represents (Fig. 2): blue = latent-recurrent, teal = value-equivalent, violet = autoregressive-token, orange = diffusion, green = JEPA, pink = video-foundation, gray = foundational or cross-cutting. A node grouping several concurrent works may list systems from other families. The field progressed from data-efficient control (PILCO) and the "training in a dream" idea (World Models, 2018), through general latent-imagination agents (DreamerV3) and value-equivalent planners (MuZero), to internet-scale interactive simulators (Genie 3, Cosmos) and world-action models (2025 to 2026). Years mark the approximate period in which each grouped set of milestones first appeared publicly (preprint or blog); several were later published in the venues listed in the comparison tables (*e.g.*DreamerV3 in *Nature* in 2025).

TABLE II: Latent-recurrent (RSSM / Dreamer-lineage) world models. "Decoder" indicates whether pixels are reconstructed. AC = actor-critic learned in imagination; MPC = model-predictive control; TD = temporal-difference value learning.

Method	Latent	Decoder	Behavior	Headline result	Venue/Year
PlaNet [13]	Gaussian+det. (RSSM)	✓	MPC (CEM)	∼200× fewer episodes vs. model-free	ICML 2019
Dreamer [14]	RSSM	✓	AC (analytic grad.)	Best-in-class on 20 DMControl visual tasks	ICLR 2020
DreamerV2 [15]	Categorical	✓	AC	First human-level Atari via WM	ICLR 2021
DreamerV3 [4]	Categorical	✓	AC	150+ tasks, one config; Minecraft diamond	Nature 2025
DayDreamer [54]	Categorical	✓	AC	Real quadruped walks in ∼1 h	CoRL 2022
Director [55]	Categorical	✓	Hierarchical AC	Solves sparse-reward 3D mazes	NeurIPS 2022
Dreaming [56]	Contrastive	×	AC	Avoids object-vanishing	ICRA 2021
MWM [57]	Masked autoenc. (ViT)	✓	AC	81.7% vs. 67.9%, 50 Meta-World tasks	CoRL 2022
Newt [62]	Reward/value (lang.-cond.)	×	MPC + TD value	1 agent, 200 tasks; MMBench	ICLR 2026
TD-MPC [60]	Reward/value	×	MPC + TD value	Strong DMControl/Meta-World efficiency	ICML 2022
TD-MPC2 [61]	Reward/value	×	MPC + TD value	1 agent, 80 tasks, multi-embodiment	ICLR 2024
Dreamer 4 [63]	Transformer+diffusion	✓	Offline AC	Offline Minecraft diamond, 100× less data	arXiv 2025

consumes, namely reward, value, and policy. Two models are value equivalent if they induce the same value estimates, and under a limited capacity budget an agent is better off spending that budget on value equivalence than on visual fidelity [65], [66].

Intuition: Two maps of a city can look completely different yet get you to every destination by the same routes; for navigating, they are equivalent. A value-equivalent world model is a map drawn only for decision-making: it may be useless for sightseeing (it cannot draw the streets) but perfect for planning the trip.

The idea was seeded by decision-aware objectives (VAML [67] weighted model errors by their effect on value) and by end-to-end abstract planners: the Predictron [68] and Value Prediction Networks [69] learned latent models trained purely on value/reward targets, directly anticipating MuZero. MuZero [16] is the canonical instance: it learns a representation function, a latent dynamics function, and a prediction head (value + policy), never reconstructs observations, and plans with Monte Carlo Tree Search (MCTS) over the learned latent model. It matched AlphaZero [70] on Go, chess, and shogi without being given the rules, and set a new state of the art on Atari. The Value Equivalence Principle [65] and Proper Value Equivalence [66] provide the theory explaining why such models work.

A rich family of extensions followed. Sampled MuZero [71] handles large or continuous action spaces by sampling actions rather than enumerating them. Stochastic MuZero [72] adds "chance nodes" so the planner can reason about genuine randomness (dice, shuffles). Gumbel MuZero [73] fixes a subtle flaw so that planning with very few simulations still guarantees policy improvement. MuZero Unplugged [74] unifies online, data-efficient, and fully offline RL by re-analyzing logged data with the latest model. On the sample-efficiency frontier, EfficientZero [75] added a self-supervised consistency loss forcing the imagined next latent to match the encoding of the real next state, becoming the first algorithm to exceed median human performance on the Atari 100k benchmark (∼2 hours of play); EfficientZero V2 [76] generalized this to continuous control and surpassed DreamerV3 across a broad suite under tight data budgets. UniZero [77] replaces MuZero's

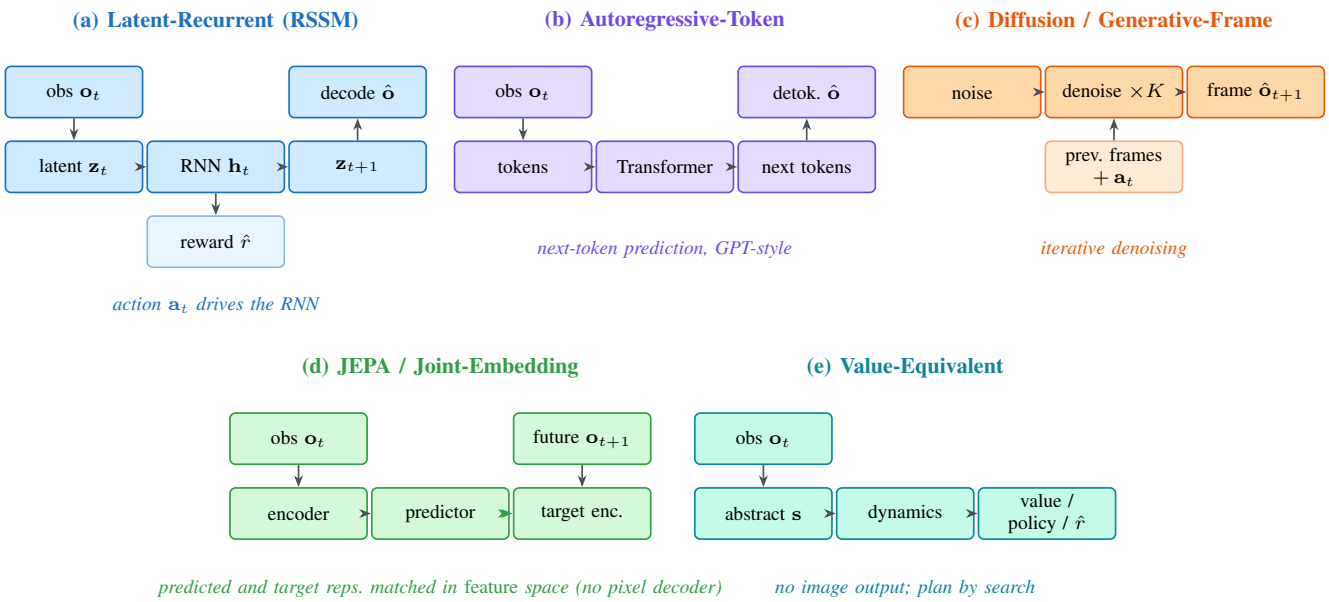

Fig. 4: The five mechanistic templates behind the six families. (a) *Latent-recurrent (RSSM)*: encode observation to a latent, roll it forward with a recurrent network, and (optionally) decode back to pixels and predict reward. (b) *Autoregressive-token*: tokenize the observation and predict the next tokens with a Transformer. (c) *Diffusion/generative-frame*: generate the next frame by iterative denoising conditioned on previous frames and the action. (d) *JEPA/joint-embedding*: predict the *representation* of the future produced by a target encoder, with no pixel decoder. (e) *Value-equivalent*: encode to an abstract state, predict only reward, value, and policy, and plan by search. This is the most abstract template and, like JEPA, produces no image. Video-foundation simulators (Section IV-F) are large-scale instances of (b) or (c).

recurrent latent dynamics with a transformer over a shared latent space, so that tree search can condition on a long history rather than a single compressed state. Its value is best read as diagnostic rather than competitive: on Atari 100k it does not beat published MuZero (human-normalized mean 0.39 against 0.56), but it markedly improves memory-demanding partially observable tasks, where the recurrent state is the bottleneck, and it trains one model across several Atari games at once. This isolates a real limitation of the classic formulation, namely that value equivalence says what to predict but leaves open how much of the past the model may condition on.

For contrast, two influential observation-reconstruction model-based methods clarify what value equivalence avoids. SimPLe [78] learns an action-conditioned video predictor of an Atari game and trains a policy inside it, pioneering the Atari 100k benchmark but spending capacity on pixels. MBPO [39] uses short "branched" rollouts from a learned dynamics ensemble, with a theoretical bound on how far the model can be trusted before compounding error dominates. MBPO thus answers the model-bias problem from a different angle: shorten the rollout rather than change the prediction target. Table III summarizes the family. Where value equivalence buys planning efficiency by refusing to draw the world at all, the next family makes the opposite bet: it reconstructs the world in full, but treats it as a sequence of tokens to be predicted like language.

C. Autoregressive-Token / Transformer World Models

This family treats the world like language. An observation is compressed into a short sequence of discrete tokens by a vector-quantized autoencoder, and a Transformer predicts the next tokens (and often reward and episode termination) conditioned on past tokens and actions. This is exactly the recipe behind large language models, transplanted to experience. Once the model can continue the "story" of a game, an agent can be trained entirely inside its imagined token rollouts.

Intuition: First learn a compact alphabet for screenshots, so each frame becomes a handful of "visual words." Then train a GPT-like model to continue the story of the game one word at a time, given the player's actions. The agent practices inside these generated stories, so it needs very little real play.

The paradigm has two roots. Trajectory Transformer [79] and Decision Transformer [80] first reframed control as token-sequence prediction, modeling states, actions, and rewards as one stream. On the world-model side, IRIS [81] converts each frame into discrete tokens with a VQ-VAE and predicts the next frame's tokens autoregressively, training a policy purely in imagination; it reached superhuman play on several Atari games from only $\sim$2 hours of experience. Δ-IRIS [82] makes this an order of magnitude faster by tokenizing only the change between frames. TWM [83] feeds states, actions, and rewards as separate tokens to a Transformer-XL for long-range memory, and STORM [84] combines DreamerV2-style categorical latents with a GPT backbone, reaching a record Atari 100k score among no-lookahead methods while training

TABLE III: Value-equivalent and planning-oriented world models, with two observation-reconstruction methods (SimPLe, MBPO) for contrast. HNS = human-normalized score on Atari 100k. MCTS = Monte Carlo Tree Search.

Method	Predicts	Planner	Headline result	Venue/Year
Predictron [68]	Value (abstract)	Internal rollouts	Accurate value prediction (mazes)	ICML 2017
VPN [69]	Reward/value	Lookahead tree	Beats DQN & obs.-prediction models	NeurIPS 2017
MuZero [16]	Reward/value/policy	MCTS	Masters Go/chess/shogi/Atari, no rules	Nature 2020
Sampled MuZero [71]	Reward/value/policy	Sampled MCTS	Continuous control (DMControl, Real-World RL)	ICML 2021
Stochastic MuZero [72]	+ chance nodes	Stochastic MCTS	SOTA 2048, backgammon	ICLR 2022
Gumbel MuZero [73]	Reward/value/policy	Gumbel search	Strong with few simulations	ICLR 2022
MuZero Unplugged [74]	Reward/value/policy	MCTS + Reanalyse	Best offline result (RL Unplugged)	NeurIPS 2021
EfficientZero [75]	+ consistency loss	MCTS	1.94 mean / 1.09 median HNS (Atari 100k)	NeurIPS 2021
EfficientZero V2 [76]	Reward/value/policy	Gumbel search	Beats DreamerV3, 50/66 tasks	ICML 2024
SimPLe [78]	*Pixels + reward*	*PPO in model*	*Pioneered Atari 100k*	*ICLR 2020*
MBPO [39]	*Next-state + reward*	*Short rollouts + SAC*	*Efficient MuJoCo control*	*NeurIPS 2019*
PETS [38]	*Next-state (ensemble)*	*MPC + uncertainty*	*$\sim$8$\times$ fewer samples than SAC (half-cheetah)*	*NeurIPS 2018*

in a few hours on a single GPU. TransDreamer [85] is the explicit bridge from the RSSM family, replacing Dreamer's recurrent core with attention.

Two results bracket what this family can be said to have learned. Othello-GPT [86] is the most-cited experimental evidence that next-token prediction alone induces a world model: a transformer trained only to predict legal moves in the board game Othello, never shown the rules or the board, turns out to carry an internal representation of board state that probing can read out, and, decisively, that intervention can edit, changing the model's subsequent move predictions in the way a genuine board change would. Prediction, on this evidence, can force structure into existence. Read alongside Vafa *et al.* [36] (Section II-B), where a model predicted orbits nearly perfectly without recovering the mechanics behind them, the pair sets the terms of the debate the rest of this survey keeps meeting: next-token prediction sometimes recovers the generating structure, and knowing when is still an open problem.

On the engineering side, tokenization granularity has itself become a research question. DeltaTok [87] pushes it to the limit: consecutive video frames share almost everything, so rather than re-encoding each frame as a grid of patch tokens it encodes only the difference between the two frames' vision-foundation-model features, as a single continuous "delta" token. Video collapses from a spatio-temporal token grid to one token per frame, a 1,024$\times$ reduction against per-patch feature grids at 512×512, and the 0.3B DeltaWorld model built on it forecasts with roughly 2,000$\times$ fewer FLOPs than Cosmos-scale generative world models. The system conditions only on past frames and a noise query, so it is a passive predictor rather than an action-conditioned world model; we include it because it shows how much of the cost in this family is spent re-encoding what did not change, and because fewer tokens per frame let a fixed context window reach much further back in time, turning a compression choice into a memory mechanism.

The family's most consequential contribution is unsupervised controllability. Google DeepMind's Genie [5] (Section V-D) trains on roughly 30k hours of unlabeled internet gameplay video (filtered from a far larger pool) and learns a small set of discrete latent actions with no action labels, so that a user can "press buttons" in a world generated from any image. This token recipe also scales to real domains: Wayve's GAIA-1 [88] (Section V-C) tokenizes driving video and predicts the next token like a language model, and Microsoft's WHAM [21] (Section V-D) tokenizes both frames and controller inputs. Open, real-time token engines such as MineWorld [89] show the approach remains competitive with diffusion for interactive world modeling, using parallel decoding for speed (frames per second, FPS, is the standard measure of interactive throughput). Table IV summarizes the family. Its strength is scalability and the natural fit with language-model tooling; its historical weakness (quantization discards fine visual detail) is precisely the opening the diffusion family exploits.

D. Diffusion and Generative-Frame World Models

Where the token family compresses the world into discrete codes, the diffusion family renders the future pixel by pixel through iterative denoising, conditioned on past frames and the agent's action. These models can act as photorealistic neural game engines: a network, rather than hand-written game code, produces each frame. Interactivity comes from action-conditioning plus autoregressive rollout: each generated frame is fed back as context for the next.

Intuition: Instead of following programmed rules, an artist who has watched thousands of hours of a game sketches, moment by moment, exactly what should appear next when you move or shoot. The central difficulty is drift: small per-frame errors accumulate over a long rollout, so the world slowly melts or "forgets" what is off-screen. Much of the research is about stopping this.

Two sub-threads exist. For control, diffusion generates whole trajectories to avoid the compounding error of step-by-step rollout. Diffuser [90] and the Decision Diffuser [91] pioneered trajectory-level diffusion planning; DWM [92] predicts a full multi-step future in one pass, yielding a $\sim$44% gain over one-step models on offline RL; PolyGRAD [93] generates whole on-policy trajectories in a single non-autoregressive pass, guiding the denoiser with the gradient of the policy's action distribution so that the synthesized states and actions stay on-policy; and DIAMOND [94] trains an RL agent inside a diffusion world model that predicts frames directly in pixel space, arguing that the visual detail discarded by token models

TABLE IV: Autoregressive-token / transformer world models. Genie appears here and is discussed further under games (Section V-D); GAIA-1 and WHAM use the same mechanism but are tabulated in their application-domain sections (Tables IX and X). UniZero is a transformer-latent MuZero variant, discussed with the value-equivalent family in Section IV-B and tabulated here for its backbone.

Method	Tokenizer	Backbone	Headline result	Venue/Year
Trajectory Transf. [79]	Discretized dims	GPT	Offline RL as sequence modeling	NeurIPS 2021
Decision Transf. [80]	Return/state/action	Causal Transformer	Return-conditioned control	NeurIPS 2021
IRIS [81]	VQ-VAE (image)	GPT	1.05 mean HNS, Atari 100k	ICLR 2023
Δ-IRIS [82]	VQ of frame-deltas	Transformer	Best Crafter score, $\sim$10$\times$ faster	ICML 2024
TWM [83]	Latent+action+reward	Transformer-XL	Competitive Atari 100k, long memory	ICLR 2023
STORM [84]	Categorical VAE	GPT	1.27 mean HNS; 4 h/game on 1 GPU	NeurIPS 2023
TransDreamer [85]	RSSM latent	Transformer (TSSM)	Long-memory 2D/3D tasks	arXiv 2022
MineWorld [89]	Image+action tokens	Transformer	Real-time (4 to 7 FPS) open Minecraft	arXiv 2025
Genie [5]	ST video tokenizer	ST-Transformer	Latent actions from unlabeled video	ICML 2024
DeltaTok [87]	$\sim$1 token per frame	Transformer	Extreme temporal compression	CVPR 2026
UniZero [77]	Per-step latent (SimNorm)	Transformer + MCTS	Long-context planning; POMDP memory	TMLR 2025
Othello-GPT [86]	*Move tokens*	*GPT*	*Editable emergent board state*	*ICLR 2023*

is decisive for control; it set a new best Atari 100k score for agents trained purely in a world model (1.46 mean HNS) and doubles as a playable Counter-Strike (CS:GO) engine.

For interactive generation, the goal is a playable simulator. GameNGen [95] was the landmark: a diffusion model (adapted Stable Diffusion) simulates DOOM at 20 FPS on a single TPU, with human raters near chance at distinguishing short clips from the real game; its key trick is noise augmentation of context frames during training, which teaches the model to correct its own drift. Oasis [96] brought a playable diffusion Minecraft to a mass audience in the browser, using a Diffusion Transformer trained with Diffusion Forcing [97]. That technique gives each frame an independent noise level, unifying autoregressive rollout and full-sequence diffusion in one model and directly attacking long-rollout stability. The Matrix [98] tackles infinite-horizon generation with a sliding-window denoising scheme, producing endlessly extendable, steerable video across game and real-world scenes. GameGen-X [99] generates and controls open-domain game video across 150+ games. DeepMind's Genie 2 and Genie 3 [17], [18] (Section V-D) are the flagship interactive foundation world models, with Genie 3 sustaining real-time 720p navigable worlds that stay coherent for minutes. Two further lines attack the failure modes that limit long interaction. To combat forgetting, WorldMem [100] equips a diffusion world model with an explicit memory bank of past frames, indexed by camera pose and timestamp and retrieved through a memory-attention mechanism, so that a scene stays consistent when the viewer looks away and later returns. To combat geometric incoherence, Geometry Forcing [101] aligns the diffusion model's internal representations with features from a geometric foundation model (a network pretrained to recover 3D structure from ordinary images), giving the generator a 3D-aware internal structure and measurably improving both visual quality and 3D consistency under camera motion and action conditioning.

Underneath all of these sits a training problem that explains drift more precisely than "errors accumulate." A model trained to predict the next frame from real previous frames is, at generation time, asked to predict from its own previous frames, which are slightly wrong; it has never practiced on its own mistakes. This mismatch is called exposure bias. Self Forcing [102] removes it directly by rolling the model out on its own generations during training, made affordable by caching. On a 1.3B model at 832×480 this converts 0.78 FPS with 103 s of latency into 17.0 FPS with 0.69 s of latency on a single H100 at equal VBench score. Self Forcing is text-to-video and has no action input, so it is not itself a world model; we include it because it is the enabling mechanism beneath the action-conditioned real-time systems that follow, and because it identifies exposure bias as the root cause of drift; in hindsight, GameNGen's noise augmentation and Diffusion Forcing's per-frame noise levels were each partially addressing that same mismatch.

Matrix-Game 3.0 [103] combines that insight with memory and reports the strongest current joint result on speed and persistence: up to 40 FPS at 720p from a 5B model while holding memory consistency over minute-long sequences, using residual prediction plus deliberate re-injection of imperfect generated frames during training, together with camera-aware retrieval from a spatial memory. RELIC [104] states the requirements as a triple that must hold simultaneously (real-time long-horizon streaming, consistent spatial memory, and precise control) and makes the attention context itself the memory: history is compressed into spatially downsampled latent tokens, each tagged with the relative action and the absolute camera pose, that stay in the key-value cache so attention can implicitly retrieve the right content when a viewer revisits a place, with no external memory bank; its 14B model runs at 16 FPS on four H100s at 480×832 for up to 20 seconds.

Table V summarizes the family. Its strength is unmatched visual fidelity and playability; its weaknesses are drift, heavy compute for real-time inference, and no guarantee that beautiful frames obey physics (Section IV-F).

E. JEPA and Joint-Embedding Predictive Architectures

The diffusion and token families predict the future in pixel or pixel-token space. The JEPA family, proposed by LeCun [28], argues this is the wrong target: high-dimensional

TABLE V: Diffusion and generative-frame world models. The upper block targets control/RL (predict trajectories or train agents in imagination); the lower block targets interactive generation (neural game engines) and the training techniques that enable it. FPS figures are as reported by the authors on the stated hardware.

Method	Use	Real-time	Headline result	Venue/Year
Diffuser [90]	Planning	×	Trajectory diffusion planning	ICML 2022
Decision Diffuser [91]	Planning	×	Return-conditioned diffusion	ICLR 2023
DWM [92]	Offline RL	×	+44% over one-step models (D4RL)	arXiv 2024
PolyGRAD [93]	On-policy RL	×	On-policy trajectory diffusion	TMLR 2024
DIAMOND [94]	RL + engine	10 FPS (CS:GO)	1.46 mean HNS, Atari 100k	NeurIPS 2024
GameNGen [95]	Neural engine	20 FPS (TPU)	Playable DOOM; PSNR 29.4	ICLR 2025
Diffusion Forcing [97]	Training paradigm	×	Stable long rollouts; guidance	NeurIPS 2024
Oasis [96]	Neural engine	20 FPS (H100)	Open, in-browser playable Minecraft	2024
The Matrix [98]	Neural engine	16 FPS	Infinite-horizon, game→real	arXiv 2024
GameGen-X [99]	Neural engine	×	Open-domain (150+ games)	ICLR 2025
Genie 2 [17]	Foundation engine	interactive	Playable 3D worlds from one image	2024
Genie 3 [18]	Foundation engine	24 FPS, 720p	Minutes-coherent, promptable worlds	2025
WorldMem [100]	Memory mechanism	×	Memory bank for long-term consistency	NeurIPS 2025
Geometry Forcing [101]	3D alignment	×	3D-consistent generation	ICLR 2026
Self Forcing [102]	Training paradigm	17.0 FPS (H100)	Removes exposure bias; 0.69 s latency	NeurIPS 2025
Matrix-Game 3.0 [103]	Neural engine	40 FPS, 720p	Minute-long memory consistency, 5B	arXiv 2026
RELIC [104]	Neural engine	16 FPS (4×H100)	Compressed camera-aware KV memory, 14B	arXiv 2025

observations contain enormous detail that is irrelevant or fundamentally unpredictable, so a good world model should predict an abstract representation of the future, not its pixels. A Joint-Embedding Predictive Architecture encodes both the input and a future/masked version through two encoders, and trains a predictor to match the target in feature space, with no pixel decoder.

Intuition: You cannot predict the exact position of every leaf blowing in the wind, but you can predict "the car ahead will slow down." JEPA learns to predict the meaning of what comes next, not its appearance, like a chess player who pictures the strategic shape of the board rather than the wood grain of each piece. The main technical risk is collapse (the model cheating by mapping everything to the same point), prevented by an asymmetric target encoder and careful regularization [105].

I-JEPA [106] was the first concrete instance: from a visible part of an image it predicts the feature representations of masked regions, beating pixel-reconstruction methods on semantic tasks while being far more compute-efficient. V-JEPA [107] extended this to video via masked spatio-temporal feature prediction, learning motion and appearance 1.5 to 6× more efficiently than pixel-generative video models. V-JEPA 2 [20] scaled to over a billion parameters and more than one million hours of internet video, yielding a general video-understanding model; crucially, its action-conditioned variant V-JEPA 2-AC post-trains a latent world model on under 62 hours of unlabeled robot video and enables zero-shot robot planning: given a goal image, it rolls out candidate actions in latent space, scores which imagined latent best matches the goal, and executes the best. Deployed on Franka robots in two different labs, it needed no environment-specific data or reward. This makes it a flagship self-supervised world-action model.

The family's 2025 to 2026 development is a shift from heuristics to theory. LeJEPA [108] replaces the accumulated tricks that kept JEPAs from collapsing (stop-gradients, teacher-student asymmetry, hand-tuned schedules) with a single principled objective. Its argument runs backward from the goal: if the point of an embedding is to support downstream prediction, one can ask which distribution of embeddings minimizes downstream risk, and the answer is the isotropic Gaussian. SIGReg (Sketched Isotropic Gaussian Regularization) then pushes embeddings toward that distribution cheaply: rather than compare full high-dimensional clouds, it casts the embeddings onto many random one-dimensional lines and checks each shadow against the shadow a Gaussian ball would cast, which costs only linear time and memory. The result is one trade-off hyperparameter, roughly fifty lines of code, validation across 10+ datasets and 60+ architectures, and 79% ImageNet-1k accuracy under a linear probe, which freezes the encoder and fits only a simple linear classifier on top so that the score grades the features rather than further training, using a large Vision Transformer (ViT-H/14) pretrained on ImageNet-1k alone.

Intuition: "Collapse" is the failure where an encoder maps every input to nearly the same vector, which makes prediction trivially easy and completely useless. Earlier JEPAs avoided it with asymmetries that worked but nobody could fully explain. LeJEPA instead specifies the shape the embedding cloud should have (a round Gaussian ball) and regularizes toward it directly.

Two consequences followed quickly. LeWM [109] shows the theory buys practical independence from foundation encoders: with only next-embedding prediction plus SIGReg, it trains stably end to end from raw pixels, which is exactly what collapse had previously prevented and what forced DINO-WM to freeze a pretrained encoder. At roughly 15M parameters it trains on a single GPU in hours and plans up to 48× faster than foundation-model-based world models while staying competitive on 2D and 3D control, reframing the field's efficiency-versus-scale trade-off. Separately, Klindt *et al.* [110] supply the guarantee the JEPA program had always lacked. Identifiability asks a sharper question than accuracy: not "does the model predict well?" but "has the

model recovered the world's actual hidden variables, up to a harmless relabeling?" They prove that alignment plus Gaussian regularization linearly recovers the world's latent variables from nonlinear observations for a broad class of worlds whose latent dimensions are mutually independent and evolve under stationary additive-noise transitions, and that the Gaussian is the unique latent distribution for which the guarantee holds. Two assumptions deserve naming, because the manipulation and driving systems of Section V violate them: transitions are stationary with additive noise, and observation is passive. Note also what the theorem does and does not license. Identifiability concerns recovery of the representation; planning additionally requires that the action-conditioned dynamics be right, which is precisely the separate problem Zhang *et al.* take up next. Zhang *et al.* [111] extend this to action-conditioned models, giving joint conditions under which a controlled world model recovers both latent states and dynamics, and quantifying how counterfactual prediction error (error on roads not taken: what would have happened had the robot pushed instead of pulled?) grows when actions fail to excite the state space in enough independent directions, meaning that a robot which only ever pushes left cannot learn what pushing up would do, no matter how much data it collects. This matters for world-action models specifically: a model can predict passively observed futures well and still be unidentifiable as a model of how actions change things.

Related work builds world models on frozen pretrained features. DINO-WM [112] learns action-conditioned dynamics directly on frozen DINOv2 patch features and plans zero-shot to image goals across six environments. HWM [113] addresses the horizon limit of single-level latent planning by stacking two world models at different time scales, so the long-horizon model's predictions serve as latent subgoals for the short-horizon model; on real Franka pick-and-place from a single goal image it reaches 70% success where single-level planning reaches 0%, and in simulated push manipulation and maze navigation it improves long-horizon success while using up to $3\times$ less planning compute. C-JEPA [114] changes what gets masked: instead of hiding image patches it hides an object's latent states across the history window, keeping only the first time step as an identity anchor that tells the model which object is missing, so the model must infer that object's state from how the other objects behave, which comes close to forcing it to learn interactions rather than appearance. This buys roughly 20% absolute improvement in counterfactual visual reasoning over the same architecture without object-level masking, and on Push-T it nearly matches patch-based DINO-WM (88.7% versus 91.3% success) while using about 1% of the latent features and planning $8\times$ faster. Structure, here, substitutes for scale. PLDM [115] trains a reconstruction-free latent dynamics model on reward-free, suboptimal offline data and shows it generalizes to unseen layouts better than model-free RL. For contrast, Navigation World Models (NWM) [116] come from the same research line but are deliberately pixel-generative (diffusion): they plan by generating viewable future observations, which sharpens the pixels-versus-representations comparison. Table VI summarizes the family. Its strengths are efficiency, transferable representations, and a natural fit for planning; its open question is whether latent prediction, by discarding appearance, also discards information a downstream task might need. That is the exact opposite of the bet made by the final family, which pushes pixel generation to the largest scale of all.

F. Video-Generation Foundation Models as Simulators

The most-discussed recent claim in the field is that if you train a video generator on enough data at enough scale, it will implicitly learn how the world works, becoming a general "world simulator" for free. OpenAI's Sora [6] crystallized this: a diffusion transformer trained to denoise "spacetime patches" of video latents exhibited emergent 3D consistency and object permanence with no hand-coded physics, and OpenAI titled its report *Video generation models as world simulators*. A wave of large models followed, most marketed as world models: Google's Veo 2/3 [117], [118] (Veo 3 adding native synchronized audio), NVIDIA's Cosmos [19] platform of open "World Foundation Models" for Physical AI (with Cosmos-Transfer1 [119] for structured sim-to-real control and Cosmos-Reason1 [120] for embodied reasoning), Google's autoregressive VideoPoet [121], Alibaba's open Wan [122], Kuaishou's Kling [123], WorldDreamer [124], and Runway's GWM-1 [125], which explicitly pivots a video-generation company toward interactive, action-conditioned world models. MAGI-1 [126] and the long-context LWM [127] pursue the streaming/long-horizon axis these simulators need. Much of this ecosystem is built on Stable Video Diffusion [128], the open-weight latent video diffusion model that codified the now-standard three-stage recipe (text-to-image pretraining, then video pretraining on a curated corpus, then high-quality video fine-tuning) and demonstrated how much of video-model quality is a data-curation problem: its Large Video Dataset began at 580M annotated clip pairs and was filtered to 152M training examples. Several driving and game world models in this survey are fine-tuned from it.

The 2026 frontier continues along two axes. Seedance 2.0 [129] is a unified native audio-video generator over four input modalities (text, image, audio, video) with dense multi-reference conditioning, producing 4 to 15 second clips at 480p and 720p; its subtitle, "Advancing Video Generation for World Complexity," states the world-modeling ambition directly. Google's Gemini Omni [130] is a natively multimodal model that accepts text, image, audio, and video and emits video, and its distinguishing feature is conversational statefulness: each instruction builds on the last while the scene stays consistent. That turns generation into an interactive editing loop, a different route to interactivity than action-conditioning, and one that makes the world model's state something a user manipulates in language.

Two releases, in 2025 and 2026, sharpened the framing considerably. Sora 2 [131] was presented explicitly as a step from video generator toward world simulator. OpenAI reports markedly better handling of buoyancy, rigidity, and contact outcomes, and generates synchronized audio jointly with the video. Real worlds have sound, so a simulator arguably should too. Cosmos 3 [7] goes further and is, in our reading, the

TABLE VI: JEPA and joint-embedding (non-generative) world models. All predict in representation space and use *no* pixel decoder, except NWM, a same-lineage pixel-generative contrast. A ✓under Action-cond. marks action-conditioned models; these also *act*, by planning in latent space, except NWM, which plans by generating pixel futures, and the theory row of Zhang *et al.*, where the mark indicates that the analysis covers action-conditioned models.

Method	Predicts in	Action-cond.	Headline result	Venue/Year
I-JEPA [106]	Image features	×	Beats MAE on semantics, compute-efficient	CVPR 2023
V-JEPA [107]	Video features	×	1.5 to 6× more efficient than pixel video	TMLR 2024
V-JEPA 2 [20]	Video features	×	1M+ hours; 77.3 SSv2, SOTA anticipation	arXiv 2025
V-JEPA 2-AC [20]	Latent (robot)	✓	Zero-shot Franka manipulation	arXiv 2025
DINO-WM [112]	Frozen DINOv2 feat.	✓	Zero-shot goal planning, 6 envs	ICML 2025
PLDM [115]	Latent	✓	Best generalization to unseen layouts	NeurIPS 2025
LeJEPA [108]	Image features	×	SIGReg; 79% ImageNet-1k linear probe	arXiv 2025
LeWM [109]	Latent (end-to-end)	✓	15M params; plans up to 48× faster	arXiv 2026
HWM [113]	Latent (2 time scales)	✓	70% vs. 0% real Franka pick-place	arXiv 2026
C-JEPA [114]	Object-level features	✓	88.7% vs. 91.3% Push-T, 8× faster	ICML 2026
Klindt et al. [110]	*(theory)*	×	*Linear identifiability; Gaussian unique*	*arXiv 2026*
Zhang et al. [111]	*(theory)*	✓	*Identifiability of controlled WMs*	*arXiv 2026*
NWM [116]	*Pixels (diffusion)*	✓	*Navigation planning by simulation*	*CVPR 2025*

clearest industrial instantiation of the world-action thesis: a single omnimodal transformer over language, images, video, audio, and action sequences. It consolidates what were previously separate vision-language models, video generators, world simulators, and world-action models into one open-weight architecture, and we take it as the most explicit industrial statement yet of the convergence thesis (Section IV-G). The caveat we applied to other vendor reports applies here too, and more sharply: the report is unreviewed, and it gives no head-to-head comparison against the separate specialist models it consolidates. Unifying modalities in one architecture is an architectural claim, not evidence that unification improves any measured capability.

Intuition: Think of a very well-read animator who, after watching an enormous amount of footage, can paint a convincing new scene from a description. The open question is whether the animator merely reproduces what it has seen, or has genuinely internalized the laws of physics.

This is where the survey must be critical. A rigorous rebuttal, PhyWorld [32], built controlled 2D physics worlds with known rules and asked whether scaled video diffusion learns them. The finding is sobering. Models fit in-distribution cases (situations resembling the training data) nearly perfectly, but they fail out-of-distribution (situations unlike anything they were trained on), with velocity errors an order of magnitude higher. They rely on "case-based" memorization of the nearest training example rather than abstracting universal laws, and scaling the models up did not fix this. A companion real-world benchmark, Physics-IQ [33], reaches the same verdict: today's video generators can produce gorgeous, realistic-looking video while getting the physics badly wrong, and visual realism does not predict physical understanding. VideoPhy-2 [132] quantifies how wide the gap remains even for recent systems: on its hard, action-centric subset the best model scores only 22% joint semantic-plus-physical adherence, with conservation of mass and momentum among the most frequently violated principles. OpenAI's own Sora report [6] candidly lists such failures (glass shattering, object-state changes). One limitation of this literature should be stated: these benchmarks evaluated 2024 and 2025 systems, and no published physics benchmark has yet been re-run on the 2026 generation (Sora 2, Veo 3, Cosmos 3, Seedance 2.0). Vendor claims of improved physics are therefore unaudited, and the benchmark verdict is firmly established only for the earlier generation. The picture is not uniformly negative, and the exceptions are informative. Garrido *et al.* [133] report that video models trained to predict in a learned representation space do acquire intuitive-physics competence, scoring above chance on violation-of-expectation tests of object permanence and shape consistency, while pixel-space video prediction and multimodal language models on the same tests score close to chance.

The strongest case for the other side is Wiedemer *et al.* [134], who argue that large video generators are becoming general-purpose vision models much as language models became general-purpose text models. Across 18,384 generated videos spanning 62 qualitative and 7 quantitative tasks, Veo 3 performs tasks it was never trained for: segmentation at 0.74 mIoU (mean intersection over union: how much the predicted region and the true region overlap, where 1.0 is perfect), edge detection at 0.77 against 0.90 for a task-specific specialist, and, most strikingly, solving 5×5 mazes at a 78% pass@10 rate, meaning a maze counts as solved if any one of ten attempts succeeds, where Veo 2 manages 14%. They name the mechanism chain-of-frames, the visual counterpart of chain-of-thought, the familiar trick of making a language model write out its intermediate steps instead of jumping straight to an answer: the model works a problem out step by step across frames, using the video itself as scratch space. The generational jump from 14% to 78% on an unrelated reasoning task is the observation that sits least comfortably with the memorization account. It is suggestive rather than decisive: the two Veo generations differ in undisclosed training data as well as in scale, so maze footage entering the newer corpus cannot be ruled out, and the memorization critique in PhyWorld concerns quantitative parameter extrapolation rather than puzzle solving, so the two literatures are not testing the same capability.

How should a reader hold these findings together? They are less contradictory than they first appear, because they measure

different things. PhyWorld and Physics-IQ ask whether a generator's outputs obey quantitative physical law out of distribution, and find that they do not. Wiedemer *et al.* ask whether the generator can perform useful visual tasks it was never trained on, and find that it can. A system can reason usefully in a domain whose governing equations it has not internalized, which is roughly true of most human intuition about physics too. Our reading is that scale is delivering broad visual competence while leaving quantitative physical extrapolation unsolved, and that this distinction, rather than a verdict for either camp, is what the evidence currently supports. On Garrido *et al.*'s reading, and on the JEPA position generally, this locates the deficit less in video as a training signal than in pixels as a prediction target. That is one of two live readings rather than a settled verdict, since it rests on a single comparison on violation-of-expectation tests. The debate therefore maps directly onto our prediction-target thread: proponents of pixel-generative simulators bet that scale yields physics; JEPA proponents [28], [133] bet that pixel prediction is the wrong objective; and value-equivalent proponents sidestep appearance entirely. Table VII summarizes the family.

G. Cross-Family Bridges and Convergence

The six families are ideal types, and the most recent systems increasingly blend them. STORM and TransDreamer bridge the RSSM and token families (categorical latents inside a Transformer). TD-MPC bridges RSSM and value-equivalent (a decoder-free latent trained on value). Dreamer 4 bridges RSSM and diffusion (a Transformer world model trained with a diffusion-style objective). Diffusion Forcing bridges autoregressive and diffusion generation. Cosmos ships both diffusion and autoregressive world-foundation models in one platform. This convergence suggests the families are best read not as competing camps but as points on a design space defined by the prediction target (pixels, tokens, latents, value), the generative mechanism (recurrence, autoregression, denoising, feature prediction), and whether the model is interactive and action-conditioned. With the modeling approaches in hand, we now turn to how they are deployed across application domains.

V. Application Domains

The same six modeling approaches are deployed, often in strikingly different guises, across six domains. This section reorganizes the landscape by application, emphasizing what each domain demands of a world model and which of the three uses of Section II-C (planning, learning-in-imagination, or data generation) dominates.

A. Reinforcement Learning and Control

This is the domain where world models were born and where sample efficiency is the currency. Here the model's quality is judged indirectly, by a policy trained inside it (Section VII-A). The dominant use is learning-in-imagination (Dreamer, IRIS, DIAMOND) and planning (MuZero, TD-MPC). Because we covered these systems by mechanism in Sections IV-A to IV-D, we do not repeat them here; the key takeaway is that on the canonical Atari 100k and DeepMind Control benchmarks, the frontier is now shared by latent-recurrent (DreamerV3), value-equivalent (EfficientZero V2), token (STORM), and diffusion (DIAMOND) models. No single approach dominates, and the best choice depends on whether visual detail, discrete randomness, or long-horizon memory is the bottleneck.

B. Robotics and World-Action Models

Robotics is where the transition from world model to world-action model is most active, driven by a brutal constraint: real robot data is slow and expensive to collect. A world model helps in all three ways (plan candidate motions, dream synthetic training data, or train a policy in imagination), and a dominant paradigm has emerged: predict the future video of the task, then act toward it.

Intuition: Tell a robot what to do; it first "daydreams" a short movie of itself doing the task from its current view, then works out which motor commands would make reality unfold like the movie. Because video is a universal interface and language composes, this can generalize to new objects and instructions with little new robot data.

The video-plan paradigm began with UniPi [135], which generates a task video with a text-conditioned diffusion model and extracts actions with an inverse-dynamics model, which works backwards from two consecutive frames to the action that must have carried the robot from the first to the second (Section VI-B). Video Language Planning [136] scales this to long horizons using tree search, in which vision-language models (VLMs) propose and score the next steps while a video model predicts what each step would look like. AVDC [137] removes the need for action labels by recovering actions from dense optical-flow correspondences in the generated video, and SuSIE [138] uses cheaper image-editing (subgoal images) instead of full video. VideoAgent [139] closes the loop, refining hallucinated plans with VLM feedback and self-improving from executed data, directly attacking the hallucination problem. RoboDreamer [140] makes the video model compositional, imagining unseen object and action combinations. GR-1 [141] and GR-2 [142] pretrain on large-scale video prediction (GR-2 uses 38M internet clips), then fine-tune to jointly predict future frames and actions. They are early generative video-language-action models; GR-2 reports $\sim$97.7% average success across more than 100 tasks. iVideoGPT [143] makes interactive video prediction efficient enough for model-based RL via compressive tokenization.

A second thread builds unified world-action models and robot foundation world models. Berkeley/Google's UniSim [144] (ICLR 2024 Outstanding Paper) is a universal action-conditioned simulator of real-world interaction; policies trained purely inside it transfer to real robots zero-shot. WorldVLA [145] unifies a VLA policy and a video world model in one autoregressive transformer, showing that jointly modeling dynamics and actions beats either alone. Unified World Models (UWM) [146] couple video and action diffusion with separate timesteps, so one network is policy, forward

TABLE VII: Large video-generation foundation models framed as world simulators, plus the two most-cited skeptical evaluations. "Action-cond." distinguishes interactive/controllable models from passive text-to-video generators.

Method	Backbone	Action-cond.	Framing / result	Venue/Year
Sora [6]	Diffusion transformer	×	"Video gen. as world simulator"	OpenAI 2024
Sora 2 [131]	Undisclosed	×	Reported better physics; audio	OpenAI 2025
Veo 2/3 [117], [118]	Diffusion	×	Realistic physics; native audio (V3)	DeepMind 2024/25
Cosmos [19]	Diffusion + AR	✓	Open WFMs for Physical AI	NVIDIA 2025
Cosmos-Transfer1 [119]	Diffusion	✓	Structured sim-to-real control	NVIDIA 2025
Cosmos-Reason1 [120]	Multimodal LLM	×	Embodied physical reasoning	NVIDIA 2025
Cosmos 3 [7]	Omnimodal transformer	✓	Unifies video, audio, and action	NVIDIA 2026
VideoPoet [121]	AR transformer	×	Unified tokenized video LLM	ICML 2024
Wan [122]	Diffusion transformer	×	Open-weight; tops VBench	Alibaba 2025
Runway GWM-1 [125]	AR (on Gen-4.5)	✓	Real-time interactive worlds	Runway 2025
LWM [127]	AR (RingAttention)	×	1M-token multimodal context	ICLR 2025
Stable Video Diffusion [128]	Latent diffusion	×	Open backbone; 3-stage recipe, LVD 580M	arXiv 2023
Seedance 2.0 [129]	Diffusion	×	Native audio-video; 4 input modalities	ByteDance 2026
Gemini Omni [130]	Omnimodal	×	Stateful conversational scene editing	Google 2026
PhyWorld [32]	*(evaluation)*	×	*Video models fail OOD physics*	*ICML 2025*
Wiedemer et al. [134]	*(evaluation)*	×	*Zero-shot tasks; "chain-of-frames"*	*arXiv 2025*
Vafa et al. [36]	*(probe)*	×	*Good prediction ≠ recovered world model*	*ICML 2025*
Physics-IQ [33]	*(evaluation)*	×	*Realism ≠ physical understanding*	*arXiv 2025*
VideoPhy-2 [132]	*(evaluation)*	×	*Best model 22% on hard subset*	*arXiv 2025*

model, inverse model, or generator, and can absorb action-free video. WHALE [147] targets two chronic weaknesses of world models, poor generalization under distribution shift and unreliable uncertainty estimates, using behavior conditioning and cheap uncertainty estimation. DreamGen [148] is the flagship data-generation use: it fine-tunes a video world model on a robot, dreams photorealistic videos of new tasks, and recovers pseudo-actions from them. Training a downstream policy (such as the open cross-embodiment VLA foundation model GR00T N1 [149][1]) on these "neural trajectories" taught a humanoid 22 new behaviors, starting from teleoperation data for only a single task. Industry world models such as the 1X World Model [150] additionally aim to evaluate policies without physical deployment, and V-JEPA 2-AC [20] (Section IV-E) provides the self-supervised, latent-planning alternative. A complementary object-centric thread [151], [152] decomposes the scene into per-object "slots" and predicts dynamics slot-by-slot, trading photorealism for compositional generalization to novel object arrangements.

The clearest recent statement of the unified view is Genie Envisioner [153], which packages the whole loop as one platform: GE-Base, an instruction-conditioned video diffusion world model of robot interaction; GE-Act, which decodes that model's latents into executable action sequences; GE-Sim, which reuses the same world model as a neural simulator for training and evaluation; and EWMBench, a benchmark scoring visual fidelity and physical consistency. A single learned model thus serves as simulator, policy, and evaluator at once.

By 2026 this thread had produced one dominant recipe, which Cosmos Policy [154] states most cleanly: post-train a pretrained video diffusion model by encoding actions, and sometimes values, as additional latent frames. Rather than bolting an action head onto a video model, it encodes actions, future state images, and values as additional latent frames (extra slots in the model's own frame sequence that carry a compressed action or value instead of a picture, so the model predicts them with exactly the machinery it already uses for images) inside the video model's own diffusion process, so a pretrained Cosmos-Predict2 becomes a robot policy through one stage of post-training with no architectural change at all. Because it predicts values as well as futures, it quietly unifies two families this survey has treated separately: it is a generative world model that also supplies exactly what a value-equivalent planner consumes (Section IV-B), and it can refine both world model and value function from its own rollouts. It reports 98.5% average success on LIBERO and 67.1% on RoboCasa, and the highest average score, 93.6 out of 100, on real-world bimanual manipulation, where the authors grade partial task progress with a rubric rather than binary success. DreamZero [155] pushes the same idea toward generalization, fine-tuning a 14B autoregressive video diffusion backbone to predict future frames and actions jointly and reporting over $2\times$ better generalization to new tasks and environments than state-of-the-art VLAs, plus a striking data result: video-only demonstrations from other robots or from humans yield over 42% relative improvement on unseen tasks from just 10 to 20 minutes of footage. Getting a 14B video model to close a control loop at 7 Hz required system-level optimization, which is itself the family's characteristic engineering problem.

A cluster of 2026 systems dissents from that recipe, and the axis on which they differ is what gets predicted rather than which backbone is used. MotuBrain [156] makes the underlying unity explicit and is the best pedagogical example in the literature of why world models and policies are two readings of one object: under a single joint distribution over video and action, one set of weights serves as policy, world model, video generator, inverse-dynamics model, or joint predictor, depending only on which conditional you sample. It reports 95.8% and 96.1% average success on RoboTwin 2.0 under clean and randomized settings and reaches 11 Hz. DSWAM [157] adds a fast/slow division of labor, pairing a fast video-co-trained executor, which like Fast-WAM below

[1]GR00T N1 is itself a VLA, not a world model; we include it only as the policy such world models serve, and in Section VIII-C as the VLA foil.

emits action chunks directly at inference without generating future video, with an optional slower vision-language planner that decomposes long tasks, and brings latency from 198 ms to 74 ms through inference optimization. PointWorld [158] changes the prediction space instead of the architecture, forecasting per-pixel 3D displacements ("point flows") from RGB-D input so that state and action share one 3D space, trained on about 2M trajectories and 500 hours and running at 0.1 s per step inside model-predictive control. WestWorld [159] drops video altogether and models low-level state-action trajectories across heterogeneous robots, routing a system-aware mixture of experts (Sys-MoE), a network split into specialist subnetworks with a small router that sends each input to only a few of them, so total capacity grows without every parameter running on every input and, separately, injecting robot physical structure as a morphological embedding on the trajectory tokens, after pretraining on 89 environments, which addresses the cross-embodiment question that the video-based systems mostly sidestep. Two further systems widen the modality: TACO [160] builds a visuo-tactile world model and uses it as a self-correcting data engine, imagining visuo-tactile futures to relabel corrective actions and improving success by 44% absolute over its base policy on contact-rich tasks, which is direct evidence against the assumption that a world model must be visual; and LPWM [161] revives object-centric modeling at real-world scale, discovering keypoints, boxes and masks from video without supervision and modeling stochastic particle dynamics through a latent action module. Across the cluster the prediction space varies (pixels, 3D point flow, low-level state, touch, particles) while the training recipe barely does, which is the clearest sign that in robotics the prediction target, not the backbone, is now the live design variable.

Two developments on the tooling side matter as much as the models. GE-Sim 2.0 [162] extends Genie Envisioner into a neural simulator built for closed-loop policy evaluation, topping the public WorldArena leaderboard at only 2B parameters and delivering a 25-frame rollout in 2.3 seconds on a single H100, and adding a state expert that decodes proprioceptive state from video latents and a world judge that scores rollouts against the task instruction to yield machine-verifiable rewards; the authors use it as an evaluator and a reward-based data filter, and name reinforcement learning inside the simulator as future work. And the action-labeling bottleneck now has a reference solution in LAPA [163], which learns discrete latent actions between consecutive frames of action-free video with a VQ-VAE, pretrains a policy to predict those latent actions, and needs only a small labeled set to decode them into real commands; it is the generalization of Genie's latent actions to robot learning and reports over 30× greater pretraining efficiency than OpenVLA while outperforming it.

These systems are measured against one dominant alternative. Diffusion Policy [164] applies the same denoising machinery as diffusion world models but to action sequences rather than future observations, with no world model at all, and became the default strong baseline on real robots (an average 46.9% relative improvement over prior methods across 12 tasks). OpenVLA [165] is the open 7B vision-language-action model trained on 970k demonstrations that maps observations to actions with no future prediction whatsoever, beating the 55B RT-2-X by 16.5 points absolute across 29 tasks. Both are made possible by Open X-Embodiment [166], the pooled dataset of 22 robots from 21 institutions demonstrating 527 skills, which established that heterogeneous robot data transfers positively and which nearly every system in this section trains or evaluates on. These are the systems a world-action model must outperform to justify the cost of predicting the future at all, and Section VIII-C returns to whether it does.

A third thread, visible mainly outside academia, tests the world-action thesis at industrial scale and along the axis of data. Dyna Robotics' Dyna-2 [3] is the most quantitatively developed statement so far. Architecturally it is a video-diffusion world-action model in which video and action are tokenized separately into two cross-attending transformer streams, with a deliberately shallower action stream that joins the video stream only in the early layers; it is trained by flow matching, with a video loss and an action loss sharing a trunk. Its distinguishing feature is the training set: more than one million hours of egocentric human video (roughly 170 years of waking experience), annotated with 3D hand-pose tracks that supply wrist poses and a continuous grasp signal in place of robot action labels. By training on nested subsets of 1k, 10k, 100k, and 1M hours, the authors report what they call the first human-to-robot transfer scaling law: held-out prediction error falls smoothly as a power law in the number of hours D (reported as $\mathrm{MSE} = 0.0691\, D^{-0.0184}$), and, more consequentially, that improvement carries across the embodiment gap onto robot data. Two different measurements should be kept apart here. The power law is fitted to held-out prediction error over the four subsets. The downstream result is four points of task success, not a fitted law: after post-training on at most ten hours of robot demonstrations per task, mean normalized performance across 14 tasks climbs $20\% \to 28\% \to 45\% \to 53\%$ along the same ladder, reported from single runs without seeds or error bars. One task, turning a key in a lockbox, stays at 0% success at every smaller scale and reaches 90% only at one million hours, which is a striking anecdote but also exactly the discontinuity a smooth scaling law does not predict. Most relevant to this survey's thesis, the authors report that video co-training is the only recipe that keeps improving as action data grows: on this evidence the world model is not scaffolding to be discarded once a policy exists.

Rhoda [2] makes the same video-first bet with a different control loop. Its Direct Video-Action (DVA) model is defined as a policy that translates the predictions of a pretrained causal video model into actions in a real-time closed loop, conditioned on a long history of robot video together with proprioception (the robot's sense of its own body: joint angles, gripper width, and forces, the machine equivalent of knowing where your hand is with your eyes shut) and language. Two design choices are instructive. First, the model carries hundreds of frames of visual context, where a typical VLA sees only a few; the authors motivate this by visual aliasing, in which two moments in a task look nearly identical but sit at very different points in the sequence, so that only memory disambiguates them. This is the same long-horizon memory

problem that Section IV-D met in generative world models, arriving here as a control requirement rather than a visual one. Second, prediction and action are separated: because the video model supplies the future, the residual problem of inferring which action produces it is small enough to be solved by an inverse-dynamics model trained on roughly ten hours of data. To keep the loop closed in real time the authors use what they call leapfrog inference, predicting far enough ahead that each prediction covers the latency of computing the next one.

Neither Dyna-2 nor Rhoda has a peer-reviewed publication, and both quantify only what they choose to; Rhoda, for instance, describes its pretraining corpus as "web-scale" without stating its size. We cite them as industrial technical reports rather than validated results, and we include them because they are the clearest evidence that the pretrain-to-imagine, fine-tune-to-act recipe (Section VI-C) has moved from research demonstration to production bet. Table VIII summarizes the domain.

C. Autonomous-Driving World Models

Driving world models predict how a traffic scene will evolve, conditioned on the ego-vehicle's actions, for three purposes: simulation (safely rehearsing rare, dangerous scenarios), data generation (synthesizing diverse labeled footage), and planning (imagining and scoring candidate maneuvers). The domain's distinctive feature is the variety of prediction spaces: multi-camera pixels, bird's-eye-view (BEV) latents, 3D semantic occupancy, and LiDAR point clouds.

MILE [168] was an early camera-only model that jointly learns a BEV latent world model and a driving policy from offline video, beating the prior CARLA state of the art by 31% in a new town. Wayve's GAIA-1 [88] scaled the token-transformer-plus-diffusion-decoder recipe to a 9B-parameter driving simulator controllable by text and action, and GAIA-2 [169] made it a controllable multi-camera latent-diffusion model with structured conditioning (ego dynamics, agents, weather, country). GigaAI's DriveDreamer [170] and DriveDreamer-2 [171] added HD-map/box grounding and a large language model (LLM) front-end that turns a plain-language request ("a car cuts in") into a generated clip. ADriver-I [172] fuses a multimodal LLM with a diffusion generator in a single vision-action loop that both predicts the next frame and outputs control. Drive-WM [173] is notable for plugging into an end-to-end planner: it imagines multiple futures for candidate maneuvers and picks the safest by an image-based reward. Call this "drive into several futures and choose." GenAD [174] and Vista [175] pushed toward large-scale, generalizable, web-data-trained driving world models with versatile action control, with Vista also serving as a training-free reward model.

The GAIA line has since scaled and, more importantly, changed purpose. GAIA-3 [176] is a 15B-parameter latent-diffusion world model trained with five times the compute and roughly ten times the data of GAIA-2, spanning nine countries across three continents, with a video tokenizer twice the size. It also puts at the center of generative simulation a formulation that dates to Chen *et al.*'s world on rails [177]: an authentic logged drive is re-simulated with a changed ego trajectory while every other agent stays pinned to what it actually did. That turns a generator into a counterfactual instrument, because the recorded future supplies ground truth for the parts of the scene the ego vehicle did not cause. GAIA-4 [178] extends this past the camera-only regime by generating camera and radar together, and reports that training specifically for the world-on-rails task improves how faithfully the model preserves the recorded world by $2.5\times$. Note the trajectory here: four generations of a driving world model moving from "can it generate plausible video?" to "can it answer what would have happened if we had steered differently?", which is the question a simulator has to answer to be useful for evaluation.

The clearest sign that general world models are becoming shared infrastructure is the Waymo World Model [179], which post-trains DeepMind's Genie 3 into a driving simulator and translates its 2D video knowledge into lidar outputs matched to Waymo's own sensor rig, with control over driving action, scene layout, and language. The strategic argument is precisely the one this survey has been tracking: general pretrained world knowledge lets the simulator stage long-tail events, a tornado or an elephant in the road, that no amount of fleet mileage would supply. No model-level metrics are published, so this remains a capability demonstration rather than a measured result. OmniDreams [180] supplies what that lacks, closing the loop in a safety-critical setting: built on the Cosmos diffusion architecture and mid-trained on 21k hours of driving, it autoregressively generates action-conditioned multi-camera video in real time coupled to a driving policy, and reports preliminary results in which a world-action model post-trained from it surpasses the VLA-based policy it was compared against, with roughly one fifth the parameters.

Beyond pixels, OccWorld [181] forecasts in 3D semantic occupancy space (predicting occupied voxels and ego motion jointly), Copilot4D [182] learns an unsupervised LiDAR point-cloud world model via discrete diffusion (cutting Chamfer distance, a standard point-cloud error measure, by $\sim 65\%$ at 1 s), WoVoGen [183] uses an explicit 4D world volume to keep multi-camera views consistent, and DriveWorld [184] reframes the world model as a 4D self-supervised pre-training objective that boosts detection, mapping, tracking, and planning. A distinct "neural digital twin" branch reconstructs editable, reactive sensor data for closed-loop testing: Waabi's UniSim [185] (a neural sensor simulator, not to be confused with the Berkeley UniSim [144]) and Tesla's disclosed neural world simulator [186] for shared FSD/Optimus training. Table IX summarizes the domain.

D. Games, Interactive Environments, and Human-Action Models

In games, a neural network replaces the game engine: it predicts the next frame given past frames and the player's actions, so the "engine" is learned from data rather than programmed. This domain produced the field's most vivid demonstrations and its clearest examples of human-action models. Early action-conditioned frame prediction in Atari [187] and recurrent environment simulators [188] presaged this line a

TABLE VIII: Robotics world models and world-action models. "Acts" marks systems that output actions (world-action models); the trailing (P)/(I)/(D) tag in the last column marks the dominant use: P(lanning), I(magination training), D(ata generation). IDM = inverse-dynamics model.

Method	Approach	Predicts	Acts	Headline result / use	Venue/Year
UniPi [135]	Diffusion video + IDM	Pixels	✓	Combinatorial generalization (P)	NeurIPS 2023
VLP [136]	Video + VLM search	Pixels	✓	Long-horizon multi-step tasks (P)	ICLR 2024
AVDC [137]	Video + flow	Pixels	✓	Acts from action-free video (P)	ICLR 2024
SuSIE [138]	Image-edit subgoals	Subgoal images	✓	SOTA CALVIN; beats RT-2-X (P)	ICLR 2024
RoboDreamer [140]	Compositional video	Pixels	✓	Plans unseen object/action combos (P)	ICML 2024
GR-1 [141]	AR video-pretrain	Pixels + actions	✓	CALVIN 88.9→94.9% (I)	ICLR 2024
GR-2 [142]	AR, 38M clips	Pixels + actions	✓	∼97.7% over 100+ tasks (I)	arXiv 2024
iVideoGPT [143]	AR token video	Pixels	×	Efficient interactive MBRL (I)	NeurIPS 2024
UniSim [144]	Diffusion simulator	Pixels	×	Zero-shot sim-to-real policies (I)	ICLR 2024
WorldVLA [145]	AR unified	Pixels + actions	✓	Joint WM+policy beats either (I)	arXiv 2025
UWM [146]	Dual diffusion	Pixels + actions	✓	One net: policy/fwd/inv/generator (I)	RSS 2025
WHALE [147]	Video (414M)	Pixels	×	Robust OOD imagination (P/I)	arXiv 2024
DreamGen [148]	Video data engine	Pixels→actions	✓	22 new humanoid behaviors (D)	arXiv 2025
V-JEPA 2-AC [20]	JEPA latent	Latent	✓	Zero-shot Franka manipulation (P)	arXiv 2025
1X World Model [150]	Video (action-cond.)	Pixels	×	Humanoid sim + policy eval (D)	2024
Genie Envisioner [153]	Video diffusion + act	Pixels + actions	✓	WM as sim, policy, and evaluator (P/I)	arXiv 2025
Dyna-2 [3]	Video diffusion + act	Pixels + actions	✓	1M h human video; scaling law (I)	Dyna 2026
Rhoda (DVA) [2]	Video predictive control	Pixels + actions	✓	Closed-loop video→action (P)	Rhoda 2026
Cosmos Policy [154]	Latent-frame act.+value	Pixels/value/act.	✓	LIBERO 98.5%, RoboCasa 67.1% (I/P)	arXiv 2026
DreamZero [155]	AR video diffusion (14B)	Pixels + actions	✓	>2× generalization vs. VLAs (I)	arXiv 2026
MotuBrain [156]	Joint video-action	Pixels + actions	✓	RoboTwin 2.0 95.8%; 11 Hz (I)	arXiv 2026
DSWAM [157]	Dual-system; no test gen.	Actions	✓	RoboTwin 2.0 92.4%; 74 ms (I)	arXiv 2026
Fast-WAM [167]	Video co-train, no test gen.	Actions	✓	LIBERO 97.6%; 190 ms vs. own baselines (I)	arXiv 2026
PointWorld [158]	3D point-flow	3D displacements	✓	2M traj.; 0.1 s in MPC (P)	CVPR 2026
WestWorld [159]	Trajectory (Sys-MoE)	Low-level states	✓	89 envs; cross-embodiment (P)	ICML 2026
TACO [160]	Visuo-tactile WM	Vision + touch	✓	+44% abs. over base policy (D)	arXiv 2026
LPWM [161]	Object-centric particles	Particle dynamics	✓	Unsup. object discovery (P/I)	ICLR 2026
GE-Sim 2.0 [162]	Video diffusion sim	Pixels	×	Tops WorldArena at 2B (D)	arXiv 2026
LAPA [163]	Latent-action pretrain	Latent actions	✓	>30× pretrain efficiency (I)	ICLR 2025
Diffusion Policy [164]	*Action diffusion*	*Actions only*	✓	*+46.9% rel. over SOTA (no WM)*	*RSS 2023*
OpenVLA [165]	*VLA (7B)*	*Actions only*	✓	*+16.5 pts vs. RT-2-X 55B (no WM)*	*CoRL 2024*

decade ago. Three frontiers recur: real-time interactivity, open-domain generation across many games, and modeling the human action stream itself.

DeepMind's Genie [5] learned controllable 2D worlds from unlabeled video via unsupervised latent actions (Section IV-C); Genie 2 [17] extended this to playable 3D worlds from a single image, and Genie 3 [18] to real-time, text-promptable, minutes-coherent worlds with editable "world events," positioned as training grounds for embodied agents such as SIMA [189]. On the diffusion side, GameNGen [95] (DOOM), DIAMOND [94] (Atari, CS:GO), and Oasis [96] (Minecraft, in-browser) demonstrated playable neural engines, while open systems MineWorld [89], Matrix-Game [190], GameFactory [191], Hunyuan-GameCraft [192], PlayGen [193], and MarioVGG [194] pursue open-domain generation, generalizable control, and reproducibility.

Three recent developments, from 2025 to 2026, move the domain forward on distinct axes. On speed and persistence together, Matrix-Game 3.0 [103] reaches 40 FPS at 720p while holding memory consistency over minute-long sequences, and RELIC [104] reaches 16 FPS with retrievable spatial memory over rollouts of up to 20 seconds (Section IV-D). On what an action is, Hunyuan-GameCraft-2 [195] adds free-form language instructions such as "open the door" or "trigger an explosion" alongside the usual keyboard and mouse signals, built on a 14B image-to-video mixture-of-experts model and evaluated with its own InterBench. That is a genuine change of interface: control extends from navigating a world to also editing it semantically, which is a different axis from the memory and speed axes and arguably a harder one, since the model must decide what a described event implies for a scene it invented. On productization, Google's Project Genie [196] packages Genie 3 into a consumer prototype that generates navigable worlds from a prompt or image, capped at 60-second generations, and a later update grounds those worlds in nearly two decades of Street View imagery [197], tying generated worlds to real places.

The most consequential result for this survey's argument, though, is SIMA 2 [198], because it closes the loop that the whole enterprise presupposes. A Gemini-grounded embodied agent bootstraps from human demonstrations, then switches to self-directed play in which Gemini proposes its own tasks and estimates its own rewards, and it improves inside Genie 3 generated worlds with no new human data. It roughly doubles SIMA 1's success rate and nearly closes the gap to human performance on training environments; on held-out MineDojo tasks it completes 26 of 50 task categories where SIMA 1 completed two. The authors present this as a first preliminary working example rather than a general result, and the self-improvement experiment is narrow, centering on navigation, but it is the first evidence we know of that an agent can self-improve in worlds imagined by another model, which is the premise on which generated environments are worth building at all. If it generalizes, the practical value of a world model

TABLE IX: Autonomous-driving world models, grouped by prediction space: pixel/BEV models above the rule, and 3D-occupancy, LiDAR, 4D, and neural-rendering models below. "Acts" marks models that also output driving actions/plans. BEV = bird's-eye view; MV = multi-view camera; FID = Fréchet Inception Distance and FVD = Fréchet Video Distance, for both of which lower is better.

Method	Approach	Prediction space	Acts	Headline result	Venue/Year
MILE [168]	RSSM	BEV latent	✓	+31% CARLA, new town	NeurIPS 2022
GAIA-1 [88]	AR token + diffusion	Pixels	×	9B; controllable rare scenarios	arXiv 2023
GAIA-2 [169]	Latent diffusion	MV pixels	×	Structured multi-country control	arXiv 2025
DriveDreamer [170]	Diffusion	Pixels	✓	First real-data driving WM	ECCV 2024
DriveDreamer-2 [171]	Diffusion + LLM	Pixels	×	FID 11.2 / FVD 55.7	AAAI 2025
ADriver-I [172]	MLLM + diffusion	Pixels	✓	Unified vision-action loop	arXiv 2023
Drive-WM [173]	Diffusion	MV pixels	✓	Plug-in planner; futures ranking	CVPR 2024
GenAD [174]	Latent diffusion	Pixels	✓	OpenDV (1700 h); zero-shot	CVPR 2024
Vista [175]	Diffusion	Pixels	×	−55% FID vs. prior driving WM	NeurIPS 2024
GAIA-3 [176]	Latent diffusion (15B)	MV pixels	×	10× data; world-on-rails	Wayve 2025
GAIA-4 [178]	Generative (GAIA line)	MV pixels + radar	×	Camera+radar; 2.5× rails fidelity	Wayve 2026
Waymo World Model [179]	Genie 3 post-trained	MV pixels + LiDAR	×	Long-tail events (tornado, animal)	Waymo 2026
OmniDreams [180]	Diffusion (Cosmos)	MV pixels	✓	Real-time closed loop; beats VLA	arXiv 2026
OccWorld [181]	AR token	3D occupancy	✓	Planning from forecast occupancy	ECCV 2024
Copilot4D [182]	Discrete diffusion	LiDAR points	×	∼65% Chamfer at 1 s	ICLR 2024
WoVoGen [183]	Diffusion	4D volume + MV	×	Consistent multi-camera video	ECCV 2024
DriveWorld [184]	Memory state-space (4D)	Occupancy latent	✓	Boosts detection/planning	CVPR 2024
UniSim (Waabi) [185]	Neural rendering	MV + LiDAR	×	Closed-loop digital twin	CVPR 2023

shifts from what it can show a person to what it can teach an agent.

The domain's signature contribution to this survey's terminology is Microsoft's Muse/WHAM [21], a "World and Human Action Model," published in *Nature*. Trained on over a billion image-action pairs (∼7 years of human play) of the game *Bleeding Edge*, WHAM tokenizes both frames and controller inputs and models them jointly, so it can generate visuals given actions, generate plausible human actions, or both. That makes it an explicit world-action model, built for creative game ideation rather than benchmark scores. Its real-time successor WHAMM [199] uses MaskGIT-style parallel decoding to run a playable neural Quake II in the browser. Table X summarizes the domain.

E. General-Purpose Simulation and Spatial Intelligence

A final, fast-emerging domain treats the world model as general infrastructure. Large video-foundation models (Sora, Cosmos, Veo; Section IV-F) are pitched as universal simulators for any downstream physical-AI task, and interactive systems such as Genie 3 and Runway GWM-1 as universal training and evaluation environments for embodied agents. A distinct branch pursues spatial intelligence: World Labs (founded by Fei-Fei Li) argues that a world model should build persistent, explorable 3D worlds rather than merely stream frames, framing spatial understanding as "AI's next frontier" [200]. A different route to the same general-infrastructure goal is Pandora [201], a language-steerable world model whose "state" is video and whose "actions" are free-form text issued at any moment. Li's later functional taxonomy splits world models into renderers, which output observations, simulators, which output physically faithful state, and planners, which output actions [30]. That scheme maps onto ours: renderers are pixel-target models (Sections IV-D and IV-F), simulators add physically faithful state, and planners are world-action models (Section VI). The unifying ambition across this domain is a reusable "world foundation model" that any agent can query: the closest computational realization yet of Craik's internal simulator.

F. Language, Code, and Agentic World Models

Every domain so far has a camera in it. Yet the definition in Section II-B says nothing about pixels: a world model is anything that predicts how an environment responds to actions. When the "environment" is a program, a web page, or a text game, the natural state is not an image but a symbolic description, and the natural model is a language model.

Intuition: A programmer does not need to picture the screen to know what a line of code will do; they simulate the program's state in their head. A code world model does exactly this, and the same idea covers any environment whose state is easier to write down than to draw.

WorldCoder [202] makes the symbolic option explicit: instead of learning a neural dynamics model, its agent interacts with an environment and writes a Python program that explains what it has seen, then plans with that program. Because the model is code, it is inspectable, and it can be moved to a related task by editing rather than retraining. The authors report better sample efficiency than deep reinforcement-learning baselines. Meta's CWM [203] scales the idea to a 32-billion-parameter open-weights language model trained not only on static source code but on observation-action execution traces: recordings of what actually happened when code ran, gathered from Python interpreters and from sandboxed containers in which an agent can issue commands and observe the results. It therefore learns to simulate step by step what code does, not only how code looks. Meta reports 65.8% pass@1 on SWE-bench Verified with test-time scaling ("pass@1" is the fraction of problems solved on the first attempt, and SWE-bench Verified is a suite of real GitHub bug-fixing tasks), a strong result for a model of this size. The framing is

TABLE X: Game and interactive-environment world models, including human-action models. "Human-act." marks systems that also model/generate the agent's action stream (world-action models). FPS as reported. Unlike Genie 1 [5], the architectures of Genie 2/3 are described only in blog posts; we classify them as autoregressive latent diffusion, consistent with Fig. 2.

Method	Approach	Game/domain	Human-act.	Headline result	Venue/Year
Genie [5]	AR token	2D platformers	×	Latent actions, no labels	ICML 2024
Genie 2 [17]	AR latent diffusion	3D from 1 image	×	Playable 3D, emergent physics	2024
Genie 3 [18]	AR latent diffusion	Text-prompted	×	720p/24 FPS, minutes, promptable	2025
GameNGen [95]	Diffusion	DOOM	×	20 FPS; near-indistinguishable	ICLR 2025
DIAMOND [94]	Diffusion	Atari, CS:GO	×	1.46 HNS; playable CS:GO	NeurIPS 2024
Oasis [96]	Diffusion (DiT)	Minecraft	×	Open, in-browser, 20 FPS	2024
Muse/WHAM [21]	AR token	Bleeding Edge	✓	1B+ action pairs; *Nature*	Nature 2025
WHAMM [199]	MaskGIT	Quake II	✓	Real-time playable (10+ FPS)	2025
MineWorld [89]	AR token	Minecraft	×	Open, real-time, action metrics	arXiv 2025
Matrix-Game [190]	Diffusion	Minecraft	×	17B; GameWorld Score benchmark	arXiv 2025
GameFactory [191]	Diffusion	Open-domain	×	Control transfers to new scenes	ICCV 2025
PlayGen [193]	Diffusion (DiT)	DOOM, Mario	×	20 FPS on RTX 2060; playability	arXiv 2024
Matrix-Game 3.0 [103]	Diffusion (5B)	Open-domain	×	40 FPS 720p; minute-long memory	arXiv 2026
RELIC [104]	Diffusion (14B)	Unreal-rendered	×	16 FPS; spatial memory	arXiv 2025
Hunyuan-GameCraft-2 [195]	Diffusion MoE (14B)	Open-domain	×	Language-instruction actions	arXiv 2025
Project Genie [196]	Genie 3 (product)	Prompted worlds	×	Street View grounding	Google 2026
SIMA 2 [198]	*Gemini agent*	*3D games; Genie 3*	✓	*Self-improves in generated worlds*	*arXiv 2025*

explicitly that of a world model: given a state (the interpreter's memory) and an action (running a line of code), predict the next state. In the same spirit, world-model-augmented (WMA) web agents [204] predict, in natural language, what a click would change on a page before committing to it.

Qwen-AgentWorld [205] scales this branch to the point where it can no longer be treated as a footnote. Where CWM predicts what code does, Qwen-AgentWorld simulates whole agentic environments across seven domains (tool calling through the Model Context Protocol (MCP), a standard interface by which a model invokes external tools, plus search, terminal, software engineering, Android, web, and operating systems), trained on more than 10M real environment-interaction trajectories and built at two mixture-of-experts scales, 35B total parameters with 3B active and 397B total with 17B active, of which the 35B model is publicly released. Its mechanism is notable for this survey's taxonomy: the model performs next-state prediction by long chain-of-thought reasoning, reasoning in language about what the environment will return rather than emitting a state vector or a frame. Set against the spectrum of Fig. 6, this is a prediction target the axis cannot hold, because a natural-language description of the next state is neither more nor less abstract than a latent, it is a different kind of thing: compositional, inspectable, and able to represent its own uncertainty in words. If simulating a terminal or a web page well enough to plan against it is achievable, the practical consequence is large, since agents could rehearse irreversible actions (deleting files, submitting forms, sending payments) before taking them.

This branch matters to the survey's central thread for two reasons. First, it adds a target that neither Fig. 6 nor the six modeling families of Fig. 2 accommodates: a program is a structured, symbolic description of dynamics rather than a vector of numbers, and unlike any point on that axis it can be executed exactly and checked. Second, it inherits a different failure profile. Pixel world models hallucinate physics; code world models instead risk writing a program that is internally coherent but wrong about the environment. That failure, at least, is easier to detect: run the program and compare its predictions against what actually happens. Will the symbolic and generative routes merge, with language models proposing structure and generative models supplying perceptual detail? That is among the most interesting open questions facing the field.

Across all six domains the recurring move is the same: from a model that merely predicts to one that also acts. We synthesize that transition from world model to world-action model next (Section VI), before turning to how such systems should be evaluated (Section VII).

VI. From Prediction to Action: World-Action Models

Sections IV and V repeatedly marked whether a system merely predicts or also acts. This section synthesizes how a predictive world model becomes a world-action model that drives the hands.

A. The Three Uses, Revisited

Recall the three uses of a world model (Section II-C). *Planning* uses the model at decision time: the agent imagines candidate action sequences and executes the best. Algorithm 1 states the generic model-predictive-control loop shared by PlaNet, TD-MPC, DINO-WM, and V-JEPA 2-AC. These methods differ only in whether the rollout happens in a compact latent or an abstract representation, and in whether the objective is a reward or a goal-matching distance. *Learning in imagination* uses the model at training time (Dreamer, IRIS, DIAMOND): a policy is trained on model-generated rollouts, so real interaction is minimized. *Data generation* uses the model offline (DreamGen, Cosmos): dreamed rollouts, relabeled with pseudo-actions, become synthetic training data.

B. How to Extract Actions

A pure video/world model predicts observations, not actions, so making it act requires one of three mechanisms.

Algorithm 1 Planning with a world model (model-predictive control).

Input: world model p_ϕ, encoder e_ψ, objective J, horizon H, current observation $\mathbf{o}_t$, goal $\mathbf{g}$
$\mathbf{z}_t \leftarrow e_\psi(\mathbf{o}_t)$ // encode to latent/representation
for each planning iteration **do**
  Sample N candidate action sequences $\mathbf{a}^{(i)}_{t:t+H}$
  for each candidate i **do**
    Roll out $\hat{\mathbf{z}}^{(i)}_{t:t+H}$ with p_ϕ under $\mathbf{a}^{(i)}_{t:t+H}$
    Score $J^{(i)}$ (predicted return, or $-\|\hat{\mathbf{z}}^{(i)}_{t+H} - e_\psi(\mathbf{g})\|$)
  end for
  Refit the sampling distribution to the top-scoring candidates (CEM/MPPI)
end for
Execute the first action $\mathbf{a}_t$ of the best sequence; observe $\mathbf{o}_{t+1}$; repeat.

(i) *Inverse dynamics*: a separate model infers the action that would carry one predicted frame to the next (UniPi [135]), or dense optical-flow correspondences recover it geometrically (AVDC [137]). (ii) *Latent actions*: an unsupervised model discovers a discrete action code from unlabeled video (Genie [5]), which a policy can then set. (iii) *Joint heads*: one network predicts both future observations and actions in a shared representation (GR-1/GR-2 [141], [142], WorldVLA [145], UWM [146]), so that modeling dynamics and choosing actions reinforce each other. (iv) *Actions as latent frames*: action and value tokens are appended to the video model's own denoising sequence, so no action head and no architectural change are needed at all (Cosmos Policy [154]); this became the default in 2026. Two refinements matter in practice. Rhoda [2] shows that once the video model supplies the future, mechanism (i) shrinks to an inverse-dynamics model trainable on roughly ten hours of data; and LAPA [163] generalizes mechanism (ii) from games to robots, learning latent actions from action-free video and decoding them with a small labeled set.

C. The Pretrained-to-Imagine, Fine-Tuned-to-Act Recipe

The recipe that came to dominate in 2025 and 2026 [23] is a two-stage process that mirrors the pretrain-then-fine-tune paradigm of language models. *Stage 1 (imagine):* pretrain a large, action-agnostic video/world model on internet-scale video so it internalizes broad visual and physical dynamics. *Stage 2 (act):* fine-tune the same model on a smaller set of action-labeled demonstrations so it also emits actions (often encoded as latent frames) and can plan at test time by generating and scoring candidate futures. NVIDIA's Cosmos-based policy, V-JEPA 2-AC, GR-2, Runway GWM-1, Dyna-2 [3], and Rhoda [2] all instantiate this pattern. The bet is that the expensive, general knowledge lives in the imagination model, and that only a thin, cheap layer of action grounding is needed on top. This is the same bet that made foundation models successful in language and vision.

Until recently the recipe rested on demonstrations rather than measurement: systems built this way worked, but nobody had shown that Stage 1 obeys a predictable return on data. Dyna-2's reported human-to-robot scaling law [3] (Section V-B) is the first direct evidence on that question, and it is favorable in the way that matters most: performance on robot tasks improves monotonically across four scales of human video, a quantity available in far greater supply than robot teleoperation. If the exponent holds, the strategic implication is large, because it converts a data problem the field cannot easily solve (robot demonstrations) into one it largely already has (egocentric video). Two cautions apply. The reported exponent is shallow, so gains come slowly in absolute terms; and the result is a single-vendor technical report on a single architecture, not yet independently replicated.

A pointed negative result complicates the recipe. Most systems above conflate two distinct things: the world model as a training-time source of representations, and the world model as a test-time imagination engine that generates futures before each action. Fast-WAM [167] separates them by keeping video co-training while deleting future generation at inference entirely, and finds that most of the benefit was in the former. It stays competitive on LIBERO (97.6%) and RoboTwin 2.0 (91.8%) without embodied pretraining, while running at 190 ms latency, over 4× faster than the imagine-then-execute baselines it evaluates against; ablations show that removing video co-training hurts far more than removing the imagined rollouts. Two qualifications matter. The evidence is a within-paper ablation rather than a cross-system win: Fast-WAM's absolute numbers sit below several systems in Table VIII that do keep test-time generation, including Cosmos Policy on LIBERO and MotuBrain on RoboTwin 2.0. And latency alone does not separate the two designs, since DSWAM reaches 74 ms and MotuBrain 11 Hz while still generating video. What Fast-WAM isolates is therefore the contribution of test-time imagination, not the speed of forgoing it.

If this holds, it revises the field's inherited intuition. The Dyna lineage (Section I) taught that the payoff of a world model is planning in imagination, and every system from PlaNet to V-JEPA 2-AC has been read in that light. Fast-WAM suggests that for contemporary video-pretrained policies, the payoff is instead that predicting the future is an excellent objective for learning representations, and that once those representations exist, rolling the model forward at decision time may be an expensive habit rather than a necessity. The counter-evidence is strong and worth stating plainly. HWM [113] reaches 70% success where single-level planning reaches 0% on real Franka pick-and-place from one goal image, which is test-time planning producing all of the performance; V-JEPA 2-AC's entire zero-shot result [20], DINO-WM [112], and PointWorld's control loop [158] likewise depend on rolling the model forward at decision time, as does Drive-WM [173] when it scores candidate maneuvers. Read together, the two sets of results suggest a boundary rather than a winner: imagination can be deleted where a policy can be trained to cover the behavior distribution in advance, and cannot where the task is specified only by a goal at test time, or where it needs multi-level subgoals. Deciding when imagination must actually be run, rather than merely trained on, strikes us as one of the more consequential open questions this survey can pose. How well any of this works, and how we would even

know, are the subject of the next section.

VII. Benchmarks and Evaluation

Evaluating a world model is hard because "good" means different things depending on how the model is used. We group the field's metrics into three families and argue that no single number suffices.

A. Sample-Efficiency Benchmarks for RL and Control

When a world model is used for control, it is judged indirectly, by the performance of a policy trained inside it under a fixed data budget. Atari 100k [78] is the canonical sample-efficiency benchmark, reported by nearly every latent, token, and diffusion world model. It allows 100k agent steps ($\sim$2 hours of play) on 26 games of the Arcade Learning Environment [206], scored by human-normalized score (HNS). The DeepMind Control Suite (DMControl) [207], built on the MuJoCo physics engine [208], is the standard for continuous control from pixels, and Crafter [209] tests broad competence via a 22-achievement tree scored by geometric mean. MineRL [210] and MineDojo [211] (Minecraft) probe long-horizon, open-ended competence, and RLBench [212] and Meta-World [213] probe manipulation; Procgen [214] isolates generalization via procedurally generated levels. Because these benchmarks are often reported from few runs, the rliable [215] methodology (interquartile mean, stratified bootstrap confidence intervals, performance profiles) is now the standard for honest aggregation.

B. Generative-Video Metrics and Their Blind Spots

When a world model generates pixels, quality is judged on the video. FVD (Fréchet Video Distance) [216] measures the distance between distributions of real and generated clips in the feature space of a video network, capturing per-frame quality and temporal coherence jointly. PSNR and SSIM [217] are per-frame reference metrics, and LPIPS [218] a learned perceptual distance; CLIPScore [219] measures text/prompt adherence. All share a critical blind spot for world models: they measure appearance, not physical correctness, action controllability, or long-horizon consistency. Worse, reference-based metrics unfairly penalize a perfectly plausible rollout that simply differs from the one recorded future. This is a serious problem for stochastic, long-horizon world models, where many futures are valid.

C. World-Model-Specific Evaluation

A wave of benchmarks targets exactly what appearance metrics miss. Physion [220] and Physion++ [221] test physical-outcome prediction against human baselines, the latter requiring online inference of hidden properties (mass, friction, elasticity). Physics-IQ [33] scores real-world physics continuation and delivers the field's sharpest finding: visual realism and physical understanding are largely uncorrelated. WorldScore [222] unifies 3D, 4D, and video world generation, scoring controllability, quality (including 3D consistency), and dynamics. VBench [223] decomposes video quality into 16 human-aligned dimensions, and VBench-2.0 [224] shifts from "superficial faithfulness" (does it look real?) to "intrinsic faithfulness" (does it obey physics and commonsense?), explicitly framing video evaluation as world-model evaluation. VideoPhy-2 [132] adds an action-centric physical-commonsense test on which even the best model reaches only 22% joint adherence.

A distinct and practically important use inverts the relationship: rather than evaluating the world model, the world model does the evaluating. WorldEval [225] uses a video world model, driven by latent actions, as a proxy for ranking real robot policies without physical deployment, and reports rankings that correlate strongly with real-world outcomes while also flagging unsafe behavior. Genie Envisioner's EWM-Bench [153] plays a similar role inside a robot world-model platform. If such proxies prove faithful, they would relieve one of the heaviest costs in robotics: evaluating policies on real hardware. Table XI summarizes both control benchmarks and evaluation metrics.

Two 2026 contributions sharpen the methodology itself. LikePhys [226] sidesteps generation quality entirely, which is the confound every generation-based physics benchmark inherits: if a model draws a scene badly, one cannot tell whether it misunderstands physics or merely renders poorly. LikePhys is training-free and instead reuses the diffusion model's own denoising objective as a stand-in for likelihood, the score a probabilistic model gives a particular video, its answer to "how expected is this?" It then asks whether the model assigns higher likelihood to a physically valid video than to a curated physically impossible counterpart. Across twelve scenarios in four physics domains its Plausibility Preference Error, the rate at which the model prefers the impossible video, aligns well with human judgment. Measuring a model's beliefs rather than its outputs is a genuinely different move, and it separates competence from rendering skill in a way the earlier benchmarks could not.

Yu *et al.* [227] then turn the critique on the field's reporting practice, and their diagnosis is uncomfortable but fair: the recurring pathology is claim/evidence mismatch, where papers assert utility their evaluations cannot establish. They document that at least six distinct things now travel under the name "world model" (action-conditioned environment models, latent imagination models, future-video predictors, and others), so that identical claims are not comparable across papers, and they propose an L0 to L7 evidential ladder running from visual plausibility up to policy-optimization utility, together with seven axes including interventional action fidelity, closed-loop rollout validity, policy-ranking agreement, model exploitability, and uncertainty calibration. We endorse the underlying discipline and note that this survey's own comparison tables are vulnerable to exactly the failure they name: a "headline result" column necessarily flattens evidence of very different strengths, which is why we have tried throughout to say what each number is measured on and to mark industrial reports as self-reported.

The overarching lesson is that robust world-model evaluation is inherently multi-axis: a system can top FVD while failing Physics-IQ, or plan superbly (high downstream reward) while producing no viewable rollout at all. Reporting should

TABLE XI: Benchmarks and evaluation for world models. Upper block: RL/control benchmarks (world model judged by downstream policy). Lower block: generative and world-model-specific evaluation (world model judged directly). Ref. = requires ground-truth future.

Benchmark / Metric	Type	What it measures	Venue/Year
Atari 100k [78]	RL benchmark	Sample-efficient play, 26 games (HNS)	ICLR 2020
DMControl [207]	RL benchmark	Continuous control from pixels	2018
Crafter [209]	RL benchmark	Breadth: 22-achievement tree	ICLR 2022
MineRL [210]	RL benchmark	Long-horizon Minecraft + demos	IJCAI 2019
MineDojo [211]	RL benchmark	Open-ended, language-specified tasks	NeurIPS 2022
Meta-World [213]	RL benchmark	Multi-task / meta manipulation	CoRL 2019
RLBench [212]	RL benchmark	100 manipulation tasks + demos	RA-L 2020
Procgen [214]	RL benchmark	Generalization (procedural levels)	ICML 2020
rliable [215]	Methodology	Honest aggregate reporting (IQM, CIs)	NeurIPS 2021
FVD [216]	Video metric	Distributional realism + coherence	2018
PSNR/SSIM [217]	Video metric (Ref.)	Per-frame pixel/structural fidelity	2004
LPIPS [218]	Video metric (Ref.)	Learned perceptual similarity	CVPR 2018
CLIPScore [219]	Alignment metric	Prompt/text adherence	EMNLP 2021
Physion(++) [220], [221]	Physics benchmark	Physical prediction vs. humans	NeurIPS 2021/23
Physics-IQ [33]	Physics benchmark	Real-world physics; realism $\neq$ understanding	arXiv 2025
WorldScore [222]	World-gen benchmark	Controllability, quality (incl. 3D consistency), dynamics	ICCV 2025
VBench(-2.0) [223], [224]	Video benchmark	16-dim. quality; physics/commonsense	CVPR 2024 / arXiv 2025
VideoPhy-2 [132]	Physics benchmark	Action-centric physical commonsense	arXiv 2025
WorldEval [225]	WM-as-evaluator	Ranks real robot policies without hardware	arXiv 2025
LikePhys [226]	Physics benchmark	Likelihood of valid vs. impossible video	ICLR 2026
Yu *et al.* [227]	Methodology	L0 to L7 evidential ladder; claim/evidence gap	arXiv 2026

therefore state the intended use and evaluate accordingly: downstream policy performance for control models; and, for generative ones, controllability plus physical and long-horizon consistency, not just per-frame realism.

VIII. Discussion and Conclusion

A. State of the Art

Synthesizing the comparison tables, the state of the art is best read per domain, because no single approach dominates everywhere.

On **sample-efficient RL** (Atari 100k; Fig. 5 collects the scores, which are spread across Tables III, IV and V), the frontier is shared across families, as Section V-A noted; the headline mean human-normalized scores make this concrete: EfficientZero $\sim$1.94, DIAMOND 1.46, STORM 1.27, with DreamerV3 also competitive. EfficientZero V2's headline claim is different in kind: breadth, winning 50 of 66 tasks across Atari 100k and both proprioceptive and visual continuous control. For **general control**, DreamerV3 [4] (150+ tasks) and TD-MPC2 [61] (80 tasks across multiple embodiments) are the reference generalists, each with a single fixed configuration, and Newt [62] pushes the count to 200 tasks trained jointly. On **robotics** (Table VIII), the 2026 frontier on the standard suites belongs to video-pretrained world-action models: Cosmos Policy reports 98.5% on LIBERO and MotuBrain 95.8% on RoboTwin 2.0, numbers high enough that these benchmarks are approaching saturation and perturbed variants (Section VIII-C) are now the more informative test. V-JEPA 2-AC remains the reference for zero-shot latent planning from a goal image, GR-2 ($\sim$97.7% over 100+ tasks) for large-scale video pretraining, and DreamGen for the data-generation route. Dyna-2 [3] reports what its authors call the first scaling law carrying from human video to robot performance, and DreamZero [155] the largest generalization margin over VLAs; both are unreviewed vendor reports, which is the point worth emphasizing. The robotics frontier is now set partly by industrial technical reports rather than peer-reviewed papers, making independent replication the domain's most pressing methodological need. On **autonomous driving** (Table IX), GAIA-3 and GAIA-4 lead multi-camera generation on scale (15B parameters, roughly ten times GAIA-2's data) and, more consequentially, on the world-on-rails counterfactual formulation; OmniDreams leads real-time closed-loop action-conditioned driving; GenAD and Vista remain the open reference points; and OccWorld and Copilot4D lead the 3D-occupancy and LiDAR prediction spaces. On **interactive generation** (Table X), Genie 3 is the leading general promptable engine, sustaining minutes of coherence at 720p, while Matrix-Game 3.0 reports the best joint speed-and-persistence numbers (40 FPS at 720p with minute-long memory consistency); GameNGen, DIAMOND, and Oasis are the landmark playable engines, and WHAM the reference human-action model.

B. Summary and Key Findings

(1) One idea, many forms. Every system in this survey is a bet on how to build Craik's internal simulator. The taxonomy's six families are not rival camps but points in a design space defined by the prediction target, the generative mechanism, and whether the model is interactive. Recent systems increasingly blend them (Section IV-G).

(2) The prediction-target debate is the field's fault line. What a world model should predict (pixels, latents, abstract representations, or value) is the deepest open question (Fig. 6). Pixel generation buys fidelity and playability at the cost of compute spent modeling unpredictable detail; value equivalence buys planning efficiency at the cost of any viewable rollout; JEPA bets that representation-space prediction captures

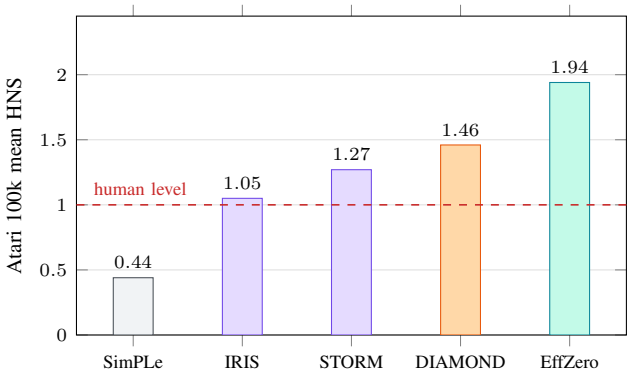

Fig. 5: Progress on the Atari 100k sample-efficiency benchmark (mean human-normalized score; 1.0 = human, dashed line), color-coded by family: gray = pixel-reconstruction (SimPLe), violet = autoregressive-token (IRIS, STORM), orange = diffusion (DIAMOND), teal = value-equivalent (EfficientZero). All scores come from roughly two hours of gameplay (100k agent steps), and bars are ordered by score, not by date. World-model agents climbed from sub-human (SimPLe, 2020) to nearly twice the human mean, with no single family dominating; notably, the value-equivalent EfficientZero (2021) still tops the metric against later token and diffusion methods. Values as reported by each paper.

exactly the actionable structure. The evidence is genuinely mixed, and the answer likely depends on the use.

(3) Realism is not understanding. The most important empirical finding of 2024 and 2025 is that visual realism and physical correctness are largely uncorrelated [32], [33], though it is established on that generation of systems and has not yet been re-tested on the 2026 models. A model can generate photorealistic video while failing basic physics out-of-distribution. This tempers the "scaling video gives a world simulator for free" thesis and motivates action-conditioning, explicit state, and better evaluation.

(4) The shift from world model to world-action model. The frontier in 2025 and 2026 is the move from passive prediction to systems that both imagine and act, via the pretrain-to-imagine, fine-tune-to-act recipe (Section VI-C). This is where robotics, games (WHAM), and driving are converging. Whether such a system must actually run its imagination at inference, or only be trained by predicting, is now contested (Section VI-C).

(5) World models attack the data bottleneck in three ways. Planning, learning-in-imagination, and data generation are complementary; dreaming synthetic training data is the fastest-growing of the three in robotics and driving, where real data is most expensive.

(6) Human video is emerging as the scaling substrate for embodied world models. The binding constraint in robotics has always been action-labeled robot data. The most consequential development of 2026, if it replicates, is evidence that this constraint can be partly routed around: Dyna-2 [3] reports that robot task performance improves predictably with hours of egocentric human video, a human-to-robot transfer scaling law, and that video co-training keeps paying off even

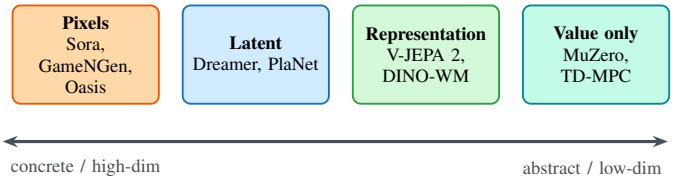

Fig. 6: The prediction-target spectrum, the field's central design axis. Moving left to right, world models predict progressively more abstract, lower-dimensional targets: raw pixels (fidelity, playability, high cost), compressed latents (efficient control), abstract representations (JEPA; action-relevant structure), and reward/value only (maximally efficient planning, no viewable rollout). Each choice trades visual fidelity against efficiency and robustness.

as action data grows. Rhoda [2] bets its whole architecture on the same premise. If this replicates, the world model stops being merely a tool for using scarce data efficiently and becomes the mechanism that makes abundant data usable at all, which is also the bottleneck the dissenting position paper of [228] identifies. The caveat is that the strongest such evidence currently comes from single-vendor reports.

C. World Models versus VLAs: The Rift in Physical AI

We now return to the industry rift previewed in Section I, which by 2026 had become the central fault line in the race to build physical AI [22]. The split runs between two camps, with a third way (below) that rejects the binary. On one side are vision-language-action (VLA) models, which are LLM-derived controllers fine-tuned to output robot actions (*e.g.*NVIDIA GR00T N1 [149] and Physical Intelligence's π models [229]). On the other are world models trained on video to predict how a scene evolves as a robot acts; these are increasingly pitched as both the simulator a robot learns from and the robot's "brain." Enthusiasm for the latter grew sharply in the mid-2020s: the video startup Luma and the humanoid company 1X each reportedly launched dedicated world-model labs, and Waymo is reported to have built a driving world model on Genie 3 [18] to rehearse rare road events such as a tornado or an animal in the road [22].

The disagreement is substantive, not merely commercial. Proponents argue a world model can internalize physics well enough to predict how an object falls and breaks, and to manufacture the simulated experience robots learn from. Skeptics counter that today's world models still hallucinate: their generated sequences are "not quite physically accurate," so training on them risks a damaging sim-to-real disconnect [22], and that this precludes high-precision, contact-rich tasks such as inserting a cable into a server tray. VLAs, meanwhile, inherit useful semantic and language understanding from their backbones, but several years in they remain brittle: small changes in object position can sharply degrade benchmark reliability, and many practitioners judge them not yet production-ready [22]. Firms deploying robots in production have taken sides on exactly these grounds, several of them reporting that they favor world models specifically because a learned

model of their own domain's physics can be built from the data their fielded robots already produce [22]. Two 2026 industrial systems raise the stakes on the world-model side of the argument: Dyna-2 [3], which reports a human-to-robot scaling law from one million hours of egocentric video, and Rhoda [2], which builds its entire control loop out of video prediction (Section V-B). Both make the video-first case in its strongest form: that the scarce resource is not model class but grounded data, and that video is where the data is.

Because much of this debate has been conducted through announcements rather than measurements, the most useful recent contribution is a controlled comparison. Zhang *et al.* [230] evaluate leading VLA policies against recently released world-action models on the LIBERO-Plus and RoboTwin 2.0-Plus benchmarks under systematic visual and language perturbations, precisely the robustness axis on which VLAs are alleged to be brittle. World-action models do hold up reasonably well: on RoboTwin 2.0-Plus the best of them reaches 74.2% success, and on LIBERO-Plus Cosmos Policy [154] reaches 82.2%. That second figure deserves to be read against the 98.5% the same model reports on unperturbed LIBERO (Section V-B). A 16-point fall under perturbation is the more instructive number, and it cuts against both camps: headline benchmark scores do not survive contact with the perturbations these suites were built to apply. But the verdict is more interesting than a win: VLAs such as $\pi_{0.5}$ reach comparable robustness on some tasks, and the authors' distinction is about cost rather than ceiling, since the VLAs need extensive training across diverse robot datasets and multiple learning objectives to get there. Read carefully, this supports neither camp's strong claim. It suggests the two model classes are converging in capability while differing in what they are expensive in, which is exactly what one would expect if, as Section IV-G argues, the families are points in a design space rather than rival species.

Two further results bear on the crux. On the world-model side, DreamZero [155] reports over $2\times$ better generalization to new tasks and environments than state-of-the-art VLAs, and OmniDreams [180] reports preliminary results in which a world-action model beats the VLA-based driving policy it was compared against; both are the kind of head-to-head the debate needed. Cutting the other way, Fast-WAM [167] finds that the advantage survives when test-time imagination is removed so long as video co-training is kept (Section VI-C). Taken together these suggest a hypothesis neither camp advertises: that what the world model contributes may be mostly representations learned by predicting rather than simulation performed while acting. We put it no more strongly than that, because the supporting result is one preprint on two simulation suites, and Section IV-E supplies clear counterexamples. If it does replicate beyond LIBERO and RoboTwin, the rift is partly a category error. A VLA trained on video prediction and a world-action model with its imagination switched off at inference are nearly the same system, and the substantive disagreement is not which model class to build but whether prediction should be an objective, a runtime procedure, or both.

A "third way" rejects the binary. Hybrid systems such as NVIDIA's Cosmos platform, and in particular Cosmos 3 [7], reason over text and images and generate physically realistic video, folding vision-language understanding, world simulation, and action generation into a single omnimodal model. A recent position paper, *Robots Need More than VLA and World Models* [228], argues that the debate over model class misses the real bottleneck: the field lacks mechanisms to convert the world's abundant unstructured behavioral data (human motion, internet video, simulation rollouts) into grounded robot supervision. Its authors propose four missing interfaces rather than a new model family: data interfaces that automatically label unstructured behavior, embodiment interfaces that retarget human motion to robot actions, world-model interfaces that supply physics-grounded 3D reasoning, and reward interfaces that infer task progress from video and language. Separately, some groups coordinate multiple copies of a language model to control a robot directly, reporting that this can outperform driving a VLA end to end [22]. The world-action model paradigm (Section VI) is itself an attempt to dissolve the rift: a single system that inherits the broad knowledge VLAs gain from pretraining while retaining the predictive, plannable dynamics a world model provides. The crux of the skeptics' case is whether prediction can be pushed to the physical precision that contact-rich manipulation demands. In our view that is the empirical question on which the debate will ultimately turn, and it motivates several of the directions below.

D. Open Challenges and Research Directions

Several challenges recur across every family and domain:

- Drift: compounding error over long autoregressive rollouts.
- Hallucination and physical/causal inconsistency: plausible but physically wrong futures.
- Memory and 3D persistence: forgetting off-screen structure.
- Controllability: faithfully obeying the conditioning action.
- Compute versus real-time interactivity.
- Honest, multi-axis evaluation.
- Safety: when a world-action model is handed autonomy.

One challenge on that list has become considerably better understood. Hallucination has usually been discussed as an architectural defect, something a better generator would eliminate. Hansen and Wang [231] argue instead that it is a data-coverage problem, and therefore predictable. They identify distinct hallucination modes and show that signals available before or during a rollout forecast when a world model is about to invent. Two mitigations follow directly: coverage-aware sampling during training, and a hallucination predictor used as a curiosity reward that steers data collection toward the regions the model does not yet cover. Reframing the problem this way matters for practice: if hallucination tracks coverage rather than capacity, then scaling the model is the wrong lever and scaling or targeting the data is the right one, and a system can be built to know when not to trust its own dream. This connects the uncertainty theme of PETS [38] and WHALE [147] to the

generative world models of Section IV-D, and it is the most actionable of the recent cross-cutting results.

We close with seven concrete research directions, ordered by importance.

(1) Physically grounded prediction. The realism-versus-understanding gap [32], [33] is the field's central scientific challenge, and the evidence in Section IV-F suggests it is a gap in quantitative extrapolation rather than in broad visual competence. Promising routes include physics-informed objectives, explicit 3D/occupancy state (OccWorld [181]), structured and object-centric latents (LPWM [161], C-JEPA [114]), and hybrids with learned or classical simulators. The goal is extrapolation to unseen configurations, not interpolation among seen ones. The concrete next experiment is the one PhyWorld made possible and nobody has yet run at scale: hold the training distribution fixed, vary only the physical parameter a model must extrapolate over, and measure error as a function of distance from the training support, family by family. Until that curve is published for pixel, latent, and representation targets alike, the field is arguing about a quantity it has not measured.

(2) Long-horizon coherence: drift, memory, and 3D persistence. These were long treated as three problems and are better read as one, because all three are ways a rollout stops agreeing with itself over time. On drift, Self Forcing [102] identifies exposure bias as the root cause and attacks it directly by training on the model's own generations, which reframes the partial remedies that preceded it (Diffusion Forcing [97], noise augmentation [95], full-trajectory diffusion [92], sliding windows [98]). On memory, the open question is now where history should live: in an external bank indexed by pose and time (WorldMem [100]), in a compressed attention context (RELIC [104]), or in explicit camera-aware retrieval (Matrix-Game 3.0 [103]). On geometry, aligning a generator's internal features with 3D structure helps (Geometry Forcing [101]), and worlds that stay coherent over long exploration will likely need 3D-native representations (World Labs [200]). A single mechanism that delivers stable, unbounded, interactive rollout remains the gating problem for playable world models.

(3) Resolving the prediction-target question empirically. This is the survey's organizing question and it is still settled more by conviction than by measurement. Head-to-head comparisons of pixel, latent, representation, and value targets on the same tasks (as NWM [116] versus V-JEPA 2-AC [20] begin to do) would replace ideology with evidence. The theory side has moved faster than the empirical side: identifiability results now say when a latent world model can be recovered at all [110], [111], which gives the comparison something firmer to stand on. What is missing is a shared harness in which the prediction target is the only variable, with the encoder, data, compute budget, and planner held fixed. The likely outcome is not a winner but a mapping from task property to target: contact-rich manipulation may need appearance that navigation can discard. Publishing that mapping would be more useful than another system paper.

(4) Unified world-action models, and controllability without action labels. Jointly training dynamics and action in one network (WorldVLA [145], UWM [146], Genie Envisioner [153], Cosmos Policy [154]) shows mutual reinforcement, and Cosmos 3 [7] scales the idea to an omnimodal model spanning language, video, audio, and action. The binding constraint is action labels, which is why unsupervised latent actions (Genie [5], LAPA [163]) matter so much: they unlock action-free internet and human video, and extending robust latent-action discovery to 3D, real-world, and long-horizon settings would decouple controllability from expensive teleoperation. Two open disagreements sit underneath. A dissenting view holds that the missing pieces are not bigger unified models but interfaces that turn unstructured human data into grounded robot supervision [228]; and Fast-WAM [167] raises the sharper possibility that prediction should be a training objective rather than a runtime procedure (Section VI-C). Establishing when imagination must actually be run, and not merely trained on, would settle both.

(5) Honest evaluation, including world models as evaluators. Evaluation must become multi-axis and physics-, action-, and consistency-aware by default (WorldScore [222], VBench-2.0 [224], Physics-IQ [33]), with honest aggregation [215]; Yu *et al.* [227] give a concrete ladder for matching claims to evidence, and LikePhys [226] shows that a model's beliefs can be probed without grading its rendering. The inverse use is equally consequential: a world model that ranks policies without physical deployment (1X [150], WorldEval [225], EWMBench [153], GE-Sim 2.0 [162]) or ranks candidate maneuvers before executing them (Drive-WM [173]) would remove the heaviest cost in robotics and driving, provided its rankings transfer. A shared, cross-domain world-model benchmark suite is still missing. The harder problem is calibration rather than coverage: a proxy evaluator is only useful if one knows in advance when to believe it, and no current work establishes that boundary. Reporting where a world-model evaluator's rankings stop agreeing with hardware would be worth more than another point of correlation.

(6) Real-time, efficient inference. Playable and deployable world models need frames in tens of milliseconds, and this has shifted from an engineering footnote to a determinant of what the field can even attempt: closing a control loop at 7 Hz with a 14B video model took dedicated systems work [155], and the cheapest route may be to spend fewer tokens per frame in the first place [87]. Advances in few-step diffusion, parallel token decoding (MineWorld [89], WHAMM [199]), efficient tokenizers, and dedicated hardware (Oasis [96]) are all part of the answer. Two framings deserve separating. Interactive generation needs low latency, so a viewer never waits; control needs a predictable deadline, since a late action is a wrong action. Systems built for the first do not automatically satisfy the second, and reporting latency without its variance hides exactly the failure that matters for a robot.

(7) Safety and reliability. As world-action models gain autonomy, new risks arise: goal misgeneralization, reward hacking against a learned model, and hallucinated rollouts driving unsafe actions. The most encouraging recent result is that hallucination appears predictable from data coverage rather than intrinsic to the architecture [231], which suggests a system can be built to know when to distrust its own dream. Uncertainty-aware world models (PETS [38], WHALE [147]),

guardrails, and verification of dreamed plans against reality are essential before deployment in safety-critical settings. The specific hazard the world-action framing introduces is that a single model supplies both the plan and the standard the plan is judged against, so an error in the model cannot be caught by consulting the model. Independent verification, whether by a classical simulator, a physical check, or a second model trained differently, is therefore not a refinement but a requirement. This is the least developed of the seven directions relative to its importance.

E. Conclusion

World models turn Craik's 1943 intuition, that intelligence rests on an internal simulator of reality, into an engineering discipline. In under a decade the field has progressed from training an agent inside a learned dream of a racing game, to internet-scale simulators that generate playable worlds from a sentence, to world-action models that both imagine the future and act to reach it. Beneath this variety lies a single organizing question, which we have used to structure the entire survey: what should a world model predict, and should it also drive the hands? The six modeling families answer the first half differently; the shift to world-action models answers the second. The hard problems that remain (physical consistency, long-horizon coherence, faithful controllability, honest evaluation, and safety) are precisely the questions that will decide which of these bets pays off, and whether learned world models become the substrate of general, embodied intelligence.

IX. Glossary of Abbreviations

AC	Actor-Critic (also Action-Conditioned).
AR	Autoregressive.
BEV	Bird's-Eye View.
CEM	Cross-Entropy Method (a sampling-based planning optimizer).
DiT	Diffusion Transformer.
DQN	Deep Q-Network.
DVA	Direct Video-Action (model).
FID	Fréchet Inception Distance.
FLOPs	Floating-Point Operations (a measure of compute cost).
FPS	Frames Per Second.
FVD	Fréchet Video Distance.
GAN	Generative Adversarial Network.
HNS	Human-Normalized Score.
IDM	Inverse-Dynamics Model.
IQM	Interquartile Mean.
JEPA	Joint-Embedding Predictive Architecture.
KV	Key-Value (as in the attention key-value cache).
LLM	Large Language Model.
LPIPS	Learned Perceptual Image Patch Similarity.
MAE	Masked Autoencoder.
MBRL	Model-Based Reinforcement Learning.
MCP	Model Context Protocol.
MCTS	Monte Carlo Tree Search.
MDP	Markov Decision Process.
mIoU	Mean Intersection over Union (a segmentation-overlap score).
MLLM	Multimodal Large Language Model.
MoE	Mixture of Experts.
MPC	Model-Predictive Control.
MPPI	Model Predictive Path Integral (a sampling-based planner).
MSE	Mean Squared Error.
OOD	Out-of-Distribution.
POMDP	Partially Observable Markov Decision Process.
PPO	Proximal Policy Optimization.
PSNR	Peak Signal-to-Noise Ratio.
RGB-D	Color image plus per-pixel Depth.
RL	Reinforcement Learning.
RNN	Recurrent Neural Network (GRU = Gated Recurrent Unit).
RSSM	Recurrent State-Space Model.
SAC	Soft Actor-Critic.
SIGReg	Sketched Isotropic Gaussian Regularization.
SOTA	State of the Art.
SSIM	Structural Similarity Index Measure.
ST	Spatio-Temporal (as in ST-Transformer).
TD	Temporal Difference (learning).
TPU	Tensor Processing Unit.
TSSM	Transformer State-Space Model.
VAE	Variational Autoencoder.
VLA	Vision-Language-Action (model).
VLM	Vision-Language Model.
VQ	Vector Quantization (VQ-VAE, VQGAN).
WAM	World-Action Model.
WFM	World Foundation Model.
WM	World Model.
WMA	World-Model-Augmented (web agents).

Acknowledgements

The author thanks the broader robot-learning and generative-modeling communities whose open publications, technical reports, and public demonstrations made this survey possible.

Declaration on the Use of AI Tools

AI assistants were used in preparing this survey, under the author's direction and following author-specified writing guidelines. Their role covered locating candidate source material, verifying that every cited work exists and that the quantities attributed to it match its primary source, assisting with drafting and editing, and proofreading for style and internal consistency. All research questions, the organizing ideas, the taxonomy, and the directions argued for in this work were proposed and decided by the author, who reviewed the whole manuscript and is solely responsible for its content. Verification was systematic but is not a guarantee: mismatches or mistakes may remain, and corrections are welcome.

References

[1] K. J. W. Craik, *The Nature of Explanation*. Cambridge University Press, 1943.

[2] Rhoda AI Team, "Causal video models are data-efficient robot policy learners," Rhoda AI Research blog, March 2026, https://www.rhoda.ai/research/direct-video-action.

[3] Dyna Robotics, "Dyna-2: A 1-million-hour scaling law for world-action models," Dyna Robotics technical report, August 2026, https://www.dyna.co/dyna-2.

[4] D. Hafner, J. Pasukonis, J. Ba, and T. Lillicrap, "Mastering diverse control tasks through world models," *Nature*, vol. 640, pp. 647–653, 2025, arXiv:2301.04104.

[5] J. Bruce, M. Dennis, A. Edwards, J. Parker-Holder, Y. Shi *et al.*, "Genie: Generative interactive environments," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2024, arXiv:2402.15391, Best Paper Award.

[6] T. Brooks, B. Peebles *et al.*, "Video generation models as world simulators," OpenAI Technical Report, 2024, https://openai.com/index/video-generation-models-as-world-simulators/.

[7] NVIDIA, "Cosmos 3: Omnimodal world models for physical AI," *arXiv preprint arXiv:2606.02800*, 2026.

[8] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. MIT Press, 2018.

[9] R. S. Sutton, "Dyna, an integrated architecture for learning, planning, and reacting," *ACM SIGART Bulletin*, vol. 2, no. 4, pp. 160–163, 1991.

[10] J. Schmidhuber, "Making the world differentiable: On using self-supervised fully recurrent neural networks for dynamic reinforcement learning and planning in non-stationary environments," Technical University of Munich, Tech. Rep. FKI-126-90, 1990.

[11] ——, "On learning to think: Algorithmic information theory for novel combinations of reinforcement learning controllers and recurrent neural world models," *arXiv preprint arXiv:1511.09249*, 2015.

[12] D. Ha and J. Schmidhuber, "Recurrent world models facilitate policy evolution," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018, arXiv:1803.10122.

[13] D. Hafner, T. Lillicrap, I. Fischer, R. Villegas, D. Ha, H. Lee, and J. Davidson, "Learning latent dynamics for planning from pixels," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2019, arXiv:1811.04551.

[14] D. Hafner, T. Lillicrap, J. Ba, and M. Norouzi, "Dream to control: Learning behaviors by latent imagination," in *International Conference on Learning Representations (ICLR)*, 2020, arXiv:1912.01603.

[15] D. Hafner, T. Lillicrap, M. Norouzi, and J. Ba, "Mastering Atari with discrete world models," in *International Conference on Learning Representations (ICLR)*, 2021, arXiv:2010.02193.

[16] J. Schrittwieser, I. Antonoglou, T. Hubert, K. Simonyan, L. Sifre, S. Schmitt, A. Guez, E. Lockhart, D. Hassabis, T. Graepel, T. Lillicrap, and D. Silver, "Mastering Atari, Go, chess and shogi by planning with a learned model," *Nature*, vol. 588, pp. 604–609, 2020, arXiv:1911.08265.

[17] Google DeepMind, "Genie 2: A large-scale foundation world model," Google DeepMind Blog, 2024, https://deepmind.google/blog/genie-2-a-large-scale-foundation-world-model/.

[18] ——, "Genie 3: A new frontier for world models," Google DeepMind Blog, 2025, https://deepmind.google/blog/genie-3-a-new-frontier-for-world-models/.

[19] N. Agarwal, A. Ali, M. Bala *et al.*, "Cosmos world foundation model platform for physical AI," *arXiv preprint arXiv:2501.03575*, 2025.

[20] M. Assran, A. Bardes, D. Fan, Q. Garrido, R. Howes, N. Ballas *et al.*, "V-JEPA 2: Self-supervised video models enable understanding, prediction and planning," *arXiv preprint arXiv:2506.09985*, 2025.

[21] A. Kanervisto, D. Bignell, L. Y. Wen, S. Devlin, K. Hofmann *et al.*, "World and human action models towards gameplay ideation," *Nature*, vol. 638, pp. 656–663, 2025.

[22] R. Drew, "World models vs VLAs: The rift dividing physical AI," The Information, AI Agenda Newsletter, 2026, https://www.theinformation.com/newsletters/ai-agenda/world-models-vs-vlas-rift-dividing-physical-ai.

[23] M. Reuss, "Pretrained to imagine, fine-tuned to act: The rise of world-action models," NVIDIA Technical Blog, 2026, published 15 June 2026. https://developer.nvidia.com/blog/pretrained-to-imagine-fine-tuned-to-act-the-rise-of-world-action-models/.

[24] J. Ding *et al.*, "Understanding world or predicting future? a comprehensive survey of world models," *ACM Computing Surveys*, vol. 58, no. 3, pp. 1–38, 2025, extended version: arXiv:2411.14499.

[25] Z. Zhu, X. Wang, W. Zhao *et al.*, "Is Sora a world simulator? a comprehensive survey on general world models and beyond," *arXiv preprint arXiv:2405.03520*, 2024.

[26] T. Feng, W. Wang, and Y. Yang, "A survey of world models for autonomous driving," *arXiv preprint arXiv:2501.11260*, 2025.

[27] B. Hou *et al.*, "World model for robot learning: A comprehensive survey," *arXiv preprint arXiv:2605.00080*, 2026.

[28] Y. LeCun, "A path towards autonomous machine intelligence," OpenReview, version 0.9.2, 2022, https://openreview.net/forum?id=BZ5a1r-kVsf.

[29] E. Xing, M. Deng, and J. Hou, "Critique of world model," *arXiv preprint arXiv:2507.05169*, 2025.

[30] F.-F. Li, "A functional taxonomy of world models," Substack essay, "Dr. Fei-Fei Li", 2026, https://drfeifei.substack.com/p/a-functional-taxonomy-of-world-models.

[31] X. Chen, H. Guo, S. Guo, B. Jiang, C. Shen, X. Shen, T. Xue, Y. Xue, M. Yu, W. Zhang, B. Zhao, B. Zhou, and M. Zhou, "A definition and roadmap for world models," *arXiv preprint arXiv:2607.06401*, 2026.

[32] B. Kang, Y. Yue, R. Lu, Z. Lin, Y. Zhao, K. Wang, G. Huang, and J. Feng, "How far is video generation from world model: A physical law perspective," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2025, arXiv:2411.02385.

[33] S. Motamed, L. Culp, K. Swersky, P. Jaini, and R. Geirhos, "Do generative video models understand physical principles? (Physics-IQ)," *arXiv preprint arXiv:2501.09038*, 2025.

[34] J. Richens, D. Abel, A. Bellot, and T. Everitt, "General agents contain world models," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2025, arXiv:2506.01622.

[35] S. Cifuentes, "General Agents Contain World Models, even under Partial Observability and Stochasticity," *arXiv preprint arXiv:2602.03146*, 2026, arXiv:2602.03146 [cs.AI]. [Online]. Available: https://arxiv.org/abs/2602.03146

[36] K. Vafa, P. G. Chang, A. Rambachan, and S. Mullainathan, "What has a foundation model found? using inductive bias to probe for world models," in *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, ser. Proceedings of Machine Learning Research, vol. 267. PMLR, 2025, pp. 60 727–60 747, arXiv:2507.06952.

[37] M. P. Deisenroth and C. E. Rasmussen, "PILCO: A model-based and data-efficient approach to policy search," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2011.

[38] K. Chua, R. Calandra, R. McAllister, and S. Levine, "Deep reinforcement learning in a handful of trials using probabilistic dynamics models," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 31, 2018, pp. 4759–4770, arXiv:1805.12114.

[39] M. Janner, J. Fu, M. Zhang, and S. Levine, "When to trust your model: Model-based policy optimization," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019, arXiv:1906.08253.

[40] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *International Conference on Learning Representations (ICLR)*, 2014.

[41] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2014.

[42] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[43] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations (ICLR)*, 2021.

[44] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[45] P. Esser, R. Rombach, and B. Ommer, "Taming transformers for high-resolution image synthesis," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[46] H. Chang, H. Zhang, L. Jiang, C. Liu, and W. T. Freeman, "MaskGIT: Masked generative image transformer," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[47] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[48] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," in *International Conference on Learning Representations (ICLR)*, 2021.

[49] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[50] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le, "Flow matching for generative modeling," in *International Conference on Learning Representations (ICLR)*, 2023.

[51] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2021.

[52] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby *et al.*, "DINOv2: Learning robust visual features without supervision," *Transactions on Machine Learning Research (TMLR)*, 2024.

[53] K. Friston, "The free-energy principle: A unified brain theory?" *Nature Reviews Neuroscience*, vol. 11, no. 2, pp. 127–138, 2010.

[54] P. Wu, A. Escontrela, D. Hafner, K. Goldberg, and P. Abbeel, "DayDreamer: World models for physical robot learning," in *Conference on Robot Learning (CoRL)*, 2022, arXiv:2206.14176.

[55] D. Hafner, K.-H. Lee, I. Fischer, and P. Abbeel, "Deep hierarchical planning from pixels," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2022, arXiv:2206.04114.

[56] M. Okada and T. Taniguchi, "Dreaming: Model-based reinforcement learning by latent imagination without reconstruction," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021, arXiv:2007.14535.

[57] Y. Seo, D. Hafner, H. Liu, F. Liu, S. James, K. Lee, and P. Abbeel, "Masked world models for visual control," in *Proc. Conf. Robot Learning (CoRL)*, ser. Proceedings of Machine Learning Research, vol. 205. PMLR, 2022, pp. 1332–1344, arXiv:2206.14244.

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

[59] R. Sekar, O. Rybkin, K. Daniilidis, P. Abbeel, D. Hafner, and D. Pathak, "Planning to explore via self-supervised world models," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2020.

[60] N. Hansen, X. Wang, and H. Su, "Temporal difference learning for model predictive control," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2022, arXiv:2203.04955.

[61] N. Hansen, H. Su, and X. Wang, "TD-MPC2: Scalable, robust world models for continuous control," in *International Conference on Learning Representations (ICLR)*, 2024, arXiv:2310.16828.

[62] ——, "Learning massively multitask world models for continuous control," in *Proc. Int. Conf. Learning Representations (ICLR)*, 2026, arXiv:2511.19584. [Online]. Available: https://arxiv.org/abs/2511.19584

[63] D. Hafner, W. Yan, and T. Lillicrap, "Training agents inside of scalable world models," *arXiv preprint arXiv:2509.24527*, 2025.

[64] B. Baker, I. Akkaya, P. Zhokhov, J. Huizinga, J. Tang, A. Ecoffet, B. Houghton, R. Sampedro, and J. Clune, "Video pretraining (VPT): Learning to act by watching unlabeled online videos," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[65] C. Grimm, A. Barreto, S. Singh, and D. Silver, "The value equivalence principle for model-based reinforcement learning," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020, arXiv:2011.03506.

[66] C. Grimm, A. Barreto, G. Farquhar, D. Silver, and S. Singh, "Proper value equivalence," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021, arXiv:2106.10316.

[67] A.-m. Farahmand, A. Barreto, and D. Nikovski, "Value-aware loss function for model-based reinforcement learning," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017.

[68] D. Silver, H. van Hasselt, M. Hessel, T. Schaul, A. Guez, T. Harley, G. Dulac-Arnold, D. Reichert, N. Rabinowitz, A. Barreto, and T. Degris, "The predictron: End-to-end learning and planning," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2017, arXiv:1612.08810.

[69] J. Oh, S. Singh, and H. Lee, "Value prediction network," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017, arXiv:1707.03497.

[70] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan, and D. Hassabis, "A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play," *Science*, vol. 362, no. 6419, pp. 1140–1144, 2018.

[71] T. Hubert, J. Schrittwieser, I. Antonoglou, M. Barekatain, S. Schmitt, and D. Silver, "Learning and planning in complex action spaces," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2021, arXiv:2104.06303.

[72] I. Antonoglou, J. Schrittwieser, S. Ozair, T. Hubert, and D. Silver, "Planning in stochastic environments with a learned model," in *International Conference on Learning Representations (ICLR)*, 2022.

[73] I. Danihelka, A. Guez, J. Schrittwieser, and D. Silver, "Policy improvement by planning with Gumbel," in *International Conference on Learning Representations (ICLR)*, 2022.

[74] J. Schrittwieser, T. Hubert, A. Mandhane, M. Barekatain, I. Antonoglou, and D. Silver, "Online and offline reinforcement learning by planning with a learned model," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021, arXiv:2104.06294.

[75] W. Ye, S. Liu, T. Kurutach, P. Abbeel, and Y. Gao, "Mastering Atari games with limited data," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021, arXiv:2111.00210.

[76] S. Wang, S. Liu, W. Ye, J. You, and Y. Gao, "EfficientZero V2: Mastering discrete and continuous control with limited data," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2024, arXiv:2403.00564.

[77] Y. Pu, Y. Niu, Z. Yang, J. Ren, H. Li, and Y. Liu, "UniZero: Generalized and efficient planning with scalable latent world models," *Transactions on Machine Learning Research*, vol. 2025, 2025, arXiv:2406.10667. [Online]. Available: https://openreview.net/forum?id=Gl6dF9soQo

[78] L. Kaiser, M. Babaeizadeh, P. Milos, B. Osinski, R. H. Campbell, K. Czechowski, D. Erhan, C. Finn, P. Kozakowski, S. Levine, A. Mohiuddin, R. Sepassi, G. Tucker, and H. Michalewski, "Model-based reinforcement learning for Atari," in *International Conference on Learning Representations (ICLR)*, 2020, arXiv:1903.00374.

[79] M. Janner, Q. Li, and S. Levine, "Offline reinforcement learning as one big sequence modeling problem," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021, arXiv:2106.02039.

[80] L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch, "Decision transformer: Reinforcement learning via sequence modeling," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021, arXiv:2106.01345.

[81] V. Micheli, E. Alonso, and F. Fleuret, "Transformers are sample-efficient world models," in *International Conference on Learning Representations (ICLR)*, 2023, arXiv:2209.00588.

[82] ——, "Efficient world models with context-aware tokenization," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2024, arXiv:2406.19320.

[83] J. Robine, M. Höftmann, T. Uelwer, and S. Harmeling, "Transformer-based world models are happy with 100k interactions," in *International Conference on Learning Representations (ICLR)*, 2023, arXiv:2303.07109.

[84] W. Zhang, G. Wang, J. Sun, Y. Yuan, and G. Huang, "STORM: Efficient stochastic transformer based world models for reinforcement learning," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2023, arXiv:2310.09615.

[85] C. Chen, Y.-F. Wu, J. Yoon, and S. Ahn, "TransDreamer: Reinforcement learning with transformer world models," *arXiv preprint arXiv:2202.09481*, 2022.

[86] K. Li, A. K. Hopkins, D. Bau, F. Viégas, H. Pfister, and M. Wattenberg, "Emergent world representations: Exploring a sequence model trained on a synthetic task," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2023, oral presentation (notable-top-5%). arXiv:2210.13382. [Online]. Available: https://arxiv.org/abs/2210.13382

[87] T. Kerssies, G. Berton, J. He, Q. Yu, W. Ma, D. de Geus, G. Dubbelman, and L.-C. Chen, "A frame is worth one token: Efficient generative world modeling with delta tokens," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026, arXiv:2604.04913.

[88] A. Hu, L. Russell, H. Yeo, Z. Murez, G. Fedoseev, A. Kendall, J. Shotton, and G. Corrado, "GAIA-1: A generative world model for autonomous driving," *arXiv preprint arXiv:2309.17080*, 2023.

[89] J. Guo, Y. Ye, T. He, H. Wu, Y. Jiang, T. Pearce, and J. Bian, "MineWorld: A real-time and open-source interactive world model on Minecraft," *arXiv preprint arXiv:2504.08388*, 2025.

[90] M. Janner, Y. Du, J. B. Tenenbaum, and S. Levine, "Planning with diffusion for flexible behavior synthesis," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2022.

[91] A. Ajay, Y. Du, A. Gupta, J. B. Tenenbaum, T. Jaakkola, and P. Agrawal, "Is conditional generative modeling all you need for decision-making?" in *International Conference on Learning Representations (ICLR)*, 2023.

[92] Z. Ding, A. Zhang, Y. Tian, and Q. Zheng, "Diffusion world model: Future modeling beyond step-by-step rollout for offline reinforcement learning," *arXiv preprint arXiv:2402.03570*, 2024.

[93] M. Rigter, J. Yamada, and I. Posner, "World models via policy-guided trajectory diffusion," *Transactions on Machine Learning Research (TMLR)*, 2024, arXiv:2312.08533.

[94] E. Alonso, A. Jelley, V. Micheli, A. Kanervisto, A. Storkey, T. Pearce, and F. Fleuret, "Diffusion for world modeling: Visual details matter in Atari," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2024, arXiv:2405.12399.

[95] D. Valevski, Y. Leviathan, M. Arar, and S. Fruchter, "Diffusion models are real-time game engines," in *International Conference on Learning Representations (ICLR)*, 2025, arXiv:2408.14837.

[96] Decart and Etched, "Oasis: A universe in a transformer," Project page, 2024, https://oasis-model.github.io/.

[97] B. Chen, D. Marti Monso, Y. Du, M. Simchowitz, R. Tedrake, and V. Sitzmann, "Diffusion forcing: Next-token prediction meets full-sequence diffusion," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2024, arXiv:2407.01392.

[98] R. Feng, H. Zhang, Z. Yang *et al.*, "The Matrix: Infinite-horizon world generation with real-time moving control," *arXiv preprint arXiv:2412.03568*, 2024.

[99] H. Che, X. He, Q. Liu, C. Jin, and H. Chen, "GameGen-X: Interactive open-world game video generation," in *International Conference on Learning Representations (ICLR)*, 2025, arXiv:2411.00769.

[100] Z. Xiao, Y. Lan, Y. Zhou, W. Ouyang, S. Yang, Y. Zeng, and X. Pan, "WorldMem: Long-term consistent world simulation with memory," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2025, arXiv:2504.12369.

[101] H. Wu, D. Wu, T. He, J. Guo, Y. Ye, Y. Duan, and J. Bian, "Geometry forcing: Marrying video diffusion and 3D representation for consistent world modeling," in *International Conference on Learning Representations (ICLR)*, 2026, arXiv:2507.07982.

[102] X. Huang, Z. Li, G. He, M. Zhou, and E. Shechtman, "Self forcing: Bridging the train-test gap in autoregressive video diffusion," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2025, spotlight. arXiv:2506.08009.

[103] Z. Wang, Z. Liu, J. Li, K. Huang, B. Xu, F. Kang, M. An, P. Wang, B. Jiang, Y. Wei, Y. Xietian, J. Pei, L. Hu, B. Jiang, H. Xue, Z. Wang, H. Sun, W. Li, W. Ouyang, X. He, Y. Liu, Y. Li, and Y. Zhou, "Matrix-Game 3.0: Real-time and streaming interactive world model with long-horizon memory," *arXiv preprint arXiv:2604.08995*, Apr. 2026, skywork AI. Project page: https://matrix-game-v3.github.io/. [Online]. Available: https://arxiv.org/abs/2604.08995

[104] Y. Hong, Y. Mei, C. Ge, Y. Xu, Y. Zhou, S. Bi, Y. Hold-Geoffroy, M. Roberts, M. Fisher, E. Shechtman, K. Sunkavalli, F. Liu, Z. Li, and H. Tan, "RELIC: Interactive video world model with long-horizon memory," arXiv preprint arXiv:2512.04040, Dec. 2025, adobe Research technical report, 22 pages. [Online]. Available: https://arxiv.org/abs/2512.04040

[105] A. Dawid and Y. LeCun, "Introduction to latent variable energy-based models: A path toward autonomous machine intelligence," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2024, no. 10, p. 104011, 2024, arXiv:2306.02572.

[106] M. Assran, Q. Duval, I. Misra, P. Bojanowski, P. Vincent, M. Rabbat, Y. LeCun, and N. Ballas, "Self-supervised learning from images with a joint-embedding predictive architecture," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, arXiv:2301.08243.

[107] A. Bardes, Q. Garrido, J. Ponce, X. Chen, M. Rabbat, Y. LeCun, M. Assran, and N. Ballas, "Revisiting feature prediction for learning visual representations from video (V-JEPA)," *Transactions on Machine Learning Research (TMLR)*, 2024, arXiv:2404.08471.

[108] R. Balestriero and Y. LeCun, "LeJEPA: Provable and scalable self-supervised learning without the heuristics," *arXiv preprint arXiv:2511.08544*, 2025. [Online]. Available: https://arxiv.org/abs/2511.08544

[109] L. Maes, Q. Le Lidec, D. Scieur, Y. LeCun, and R. Balestriero, "LeWorldModel: Stable end-to-end joint-embedding predictive architecture from pixels," *arXiv preprint arXiv:2603.19312*, 2026, version 3, June 2026. [Online]. Available: https://arxiv.org/abs/2603.19312

[110] D. Klindt, Y. LeCun, and R. Balestriero, "When does LeJEPA learn a world model?" *arXiv preprint arXiv:2605.26379*, May 2026. [Online]. Available: https://arxiv.org/abs/2605.26379

[111] X. Zhang, Y. Guan, B. Zhang, H. Li, Y.-Q. Zhang, and S. E. Li, "On the identifiability of controlled world models," *arXiv preprint arXiv:2607.22430*, 2026. [Online]. Available: https://arxiv.org/abs/2607.22430

[112] G. Zhou, H. Pan, Y. LeCun, and L. Pinto, "DINO-WM: World models on pre-trained visual features enable zero-shot planning," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2025, arXiv:2411.04983.

[113] W. Zhang, B. Terver, A. Zholus, S. Chitnis, H. Sutaria, M. Assran, R. Balestriero, A. Bar, A. Bardes, Y. LeCun, and N. Ballas, "Hierarchical planning with latent world models," arXiv preprint arXiv:2604.03208, 2026, version 2, June 2026. [Online]. Available: https://arxiv.org/abs/2604.03208

[114] H. Nam, Q. Le Lidec, L. Maes, Y. LeCun, and R. Balestriero, "Causal-JEPA: Learning world models through object-level latent masking," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2026, arXiv:2602.11389.

[115] V. Sobal, W. Zhang, K. Cho, R. Balestriero, T. G. J. Rudner, and Y. LeCun, "Learning from reward-free offline data: A case for planning with latent dynamics models," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2025, arXiv:2502.14819.

[116] A. Bar, G. Zhou, D. Tran, T. Darrell, and Y. LeCun, "Navigation world models," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025, arXiv:2412.03572.

[117] Google, "State-of-the-art video and image generation with Veo 2 and Imagen 3," Google Blog (The Keyword), 16 December 2024, 2024, https://blog.google/innovation-and-ai/models-and-research/google-labs/video-image-generation-update-december-2024/.

[118] Google DeepMind, "Veo 3," Google DeepMind, 2025, https://deepmind.google/models/veo/, accessed 2026-08-12.

[119] NVIDIA, "Cosmos-Transfer1: Conditional world generation with adaptive multimodal control," *arXiv preprint arXiv:2503.14492*, 2025.

[120] ——, "Cosmos-Reason1: From physical common sense to embodied reasoning," *arXiv preprint arXiv:2503.15558*, 2025.

[121] D. Kondratyuk, L. Yu, X. Gu *et al.*, "VideoPoet: A large language model for zero-shot video generation," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2024, arXiv:2312.14125.

[122] Wan Team, Alibaba Group, "Wan: Open and advanced large-scale video generative models," *arXiv preprint arXiv:2503.20314*, 2025.

[123] Kuaishou Technology, "Kling AI: Video generation model," Kuaishou, 2024, https://klingai.com/.

[124] X. Wang, Z. Zhu *et al.*, "WorldDreamer: Towards general world models for video generation via predicting masked tokens," *arXiv preprint arXiv:2401.09985*, 2024.

[125] Runway Research, "Introducing Runway GWM-1: A general world model," Runway Research, 2025, https://runwayml.com/research/introducing-runway-gwm-1.

[126] Sand AI, "MAGI-1: Autoregressive video generation at scale," *arXiv preprint arXiv:2505.13211*, 2025.

[127] H. Liu, W. Yan, M. Zaharia, and P. Abbeel, "World model on million-length video and language with blockwise RingAttention," in *International Conference on Learning Representations (ICLR)*, 2025, arXiv:2402.08268.

[128] A. Blattmann, T. Dockhorn, S. Kulal, D. Mendelevitch, M. Kilian, D. Lorenz, Y. Levi, Z. English, V. Voleti, A. Letts, V. Jampani, and R. Rombach, "Stable Video Diffusion: Scaling latent video diffusion models to large datasets," *arXiv preprint arXiv:2311.15127*, 2023.

[129] Team Seedance, D. Chen, L. Chen, X. Chen *et al.*, "Seedance 2.0: Advancing video generation for world complexity," arXiv preprint arXiv:2604.14148, Apr. 2026, seedance 2.0 Model Card. [Online]. Available: https://arxiv.org/abs/2604.14148

[130] K. Kavukcuoglu, "Introducing Gemini Omni," Google Blog (The Keyword), Google DeepMind, May 2026, announced at Google I/O 2026. Shipped variant: Gemini Omni Flash. Quantitative human-rater evaluations reported on the model page, https://deepmind.google/models/gemini-omni/. Accessed: 2026-08-12. [Online]. Available: https://blog.google/innovation-and-ai/models-and-research/gemini-models/gemini-omni/

[131] OpenAI, "Sora 2 is here," OpenAI, 2025, released 30 September 2025. https://openai.com/index/sora-2/.

[132] H. Bansal, C. Peng, Y. Bitton, R. Goldenberg, A. Grover, and K.-W. Chang, "VideoPhy-2: A challenging action-centric physical commonsense evaluation in video generation," *arXiv preprint arXiv:2503.06800*, 2025.

[133] Q. Garrido, N. Ballas, M. Assran, A. Bardes, L. Najman, M. Rabbat, E. Dupoux, and Y. LeCun, "Intuitive physics understanding emerges from self-supervised pretraining on natural videos," *arXiv preprint arXiv:2502.11831*, 2025.

[134] T. Wiedemer, Y. Li, P. Vicol, S. S. Gu, N. Matarese, K. Swersky, B. Kim, P. Jaini, and R. Geirhos, "Video models are zero-shot learners and reasoners," arXiv preprint arXiv:2509.20328, 2025. [Online]. Available: https://arxiv.org/abs/2509.20328

[135] Y. Du, S. Yang, B. Dai, H. Dai, O. Nachum, J. B. Tenenbaum, D. Schuurmans, and P. Abbeel, "Learning universal policies via text-guided video generation," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2023, arXiv:2302.00111.

[136] Y. Du, M. Yang, P. Florence, F. Xia, A. Wahid, B. Ichter, P. Sermanet, T. Yu, P. Abbeel, J. B. Tenenbaum, L. P. Kaelbling, A. Zeng, and J. Tompson, "Video language planning," in *International Conference on Learning Representations (ICLR)*, 2024, arXiv:2310.10625.

[137] P.-C. Ko, J. Mao, Y. Du, S.-H. Sun, and J. B. Tenenbaum, "Learning to act from actionless videos through dense correspondences (AVDC)," in *International Conference on Learning Representations (ICLR)*, 2024, arXiv:2310.08576.

[138] K. Black, M. Nakamoto, P. Atreya, H. Walke, C. Finn, A. Kumar, and S. Levine, "Zero-shot robotic manipulation with pre-trained image-editing diffusion models (SuSIE)," in *International Conference on Learning Representations (ICLR)*, 2024, arXiv:2310.10639.

[139] A. Soni, S. Venkataraman, A. Chandra, S. Fischmeister, P. Liang, B. Dai, and S. Yang, "VideoAgent: Self-improving video generation for embodied planning," *arXiv preprint arXiv:2410.10076*, 2024.

[140] S. Zhou, Y. Du, J. Chen, Y. Li, D.-Y. Yeung, and C. Gan, "RoboDreamer: Learning compositional world models for robot imagination," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2024, arXiv:2404.12377.

[141] H. Wu, Y. Jing, C. Cheang, G. Chen, J. Xu, X. Li, M. Liu, H. Li, and T. Kong, "Unleashing large-scale video generative pre-training for visual robot manipulation (GR-1)," in *International Conference on Learning Representations (ICLR)*, 2024, arXiv:2312.13139.

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

[143] J. Wu, S. Yin, N. Feng, X. He, D. Li, J. Hao, and M. Long, "iVideoGPT: Interactive videogpts are scalable world models," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2024, arXiv:2405.15223.

[144] M. Yang, Y. Du, K. Ghasemipour, J. Tompson, L. Kaelbling, D. Schuurmans, and P. Abbeel, "Learning interactive real-world simulators (UniSim)," in *International Conference on Learning Representations (ICLR)*, 2024, arXiv:2310.06114, Outstanding Paper Award.

[145] J. Cen *et al.*, "WorldVLA: Towards autoregressive action world model," *arXiv preprint arXiv:2506.21539*, 2025.

[146] C. Zhu, R. Yu, S. Feng, B. Burchfiel, P. Shah, and A. Gupta, "Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets," in *Proceedings of Robotics: Science and Systems (RSS)*, 2025, arXiv:2504.02792.

[147] Z. Zhang, R. Chen, J. Ye, Y. Sun, P. Wang, J. Pang, K. Li, T. Liu, H. Lin, Y. Yu, and Z.-H. Zhou, "WHALE: Towards generalizable and scalable world models for embodied decision-making," *arXiv preprint arXiv:2411.05619*, 2024.

[148] J. Jang *et al.*, "DreamGen: Unlocking generalization in robot learning through video world models," *arXiv preprint arXiv:2505.12705*, 2025.

[149] NVIDIA, "GR00T N1: An open foundation model for generalist humanoid robots," *arXiv preprint arXiv:2503.14734*, 2025.

[150] 1X Technologies, "1X world model," 1X Technologies, 2024, https://www.1x.tech/discover/1x-world-model.

[151] T. Kipf, E. van der Pol, and M. Welling, "Contrastive learning of structured world models," in *International Conference on Learning Representations (ICLR)*, 2020.

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

[153] Y. Liao, P. Zhou, S. Huang, D. Yang, S. Chen, Y. Jiang, Y. Hu, J. Cai, S. Liu, J. Luo, L. Chen, S. Yan, M. Yao, and G. Ren, "Genie Envisioner: A unified world foundation platform for robotic manipulation," *arXiv preprint arXiv:2508.05635*, 2025.

[154] M. J. Kim, Y. Gao, T.-Y. Lin, Y.-C. Lin, Y. Ge, G. Lam, P. Liang, S. Song, M.-Y. Liu, C. Finn, and J. Gu, "Cosmos Policy: Fine-tuning video models for visuomotor control and planning," *arXiv preprint arXiv:2601.16163*, 2026. [Online]. Available: https://arxiv.org/abs/2601.16163

[155] S. Ye, Y. Ge, K. Zheng, S. Gao, S. Yu, G. Kurian, S. Indupuru, Y. L. Tan, C. Zhu, J. Xiang, A. Malik, K. Lee, W. Liang, N. Ranawaka, J. Gu, Y. Xu, G. Wang, F. Hu, A. Narayan, J. Bjorck, J. Wang, G. Kim, D. Niu, R. Zheng, Y. Xie, J. Wu, Q. Wang, R. Julian, D. Xu, Y. Du, Y. Chebotar, S. Reed, J. Kautz, Y. Zhu, L. Fan, and J. Jang, "World action models are zero-shot policies," *arXiv preprint arXiv:2602.15922*, 2026, arXiv:2602.15922 [cs.RO]. Project page: https://dreamzero0.github.io/.

[156] MotuBrain Team, C. Xiang, F. Bao, H. Liu, H. Tan, H. Bi, J. Li, J. Liu, J. Pang, K. Jing, L. Liu, M. Cai, R. Cui, R. Zhao, R. Wang, S. Huang, Y. Feng, Y. Rong, Z. Wang, and J. Zhu, "MotuBrain: An advanced world action model for robot control," arXiv preprint arXiv:2604.27792, 2026. [Online]. Available: https://arxiv.org/abs/2604.27792

[157] J. Zhu, J. Zhang, T. Su, T. Liu, Z. Wang, K. Xie, Z. Huang, C. Ma, Y. He, T. Wang, H. Wang, W. Ding, and Y. Xu, "DSWAM: A dual-system world action foundation model for fine-grained robot manipulation," *arXiv preprint arXiv:2607.04927*, 2026. [Online]. Available: https://arxiv.org/abs/2607.04927

[158] W. Huang, Y.-W. Chao, A. Mousavian, M.-Y. Liu, D. Fox, K. Mo, and L. Fei-Fei, "PointWorld: Scaling 3D world models for in-the-wild robotic manipulation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2026, pp. 20 765–20 779, arXiv:2601.03782.

[159] Y. Wang, J. Kong, S. Wei, X. Li, H. Lin, H. Zhao, T. Zhou, L. Gan, and H. Shao, "WestWorld: A knowledge-encoded scalable trajectory world model for diverse robotic systems," in *Proc. 43rd Int. Conf. Mach. Learn. (ICML)*, Seoul, South Korea, 2026, spotlight. arXiv:2603.14392.

[160] S. Liu, Y. Jia, Y. Yan, J. Liu, X. Zhang, Q. Feng, Y. Guo, S. Zhou, B. Shi, and S. Zhang, "TACO: TActile world model as a Self-COrrector for scalable VLA post-training," arXiv preprint arXiv:2607.02840, Jul. 2026. [Online]. Available: https://arxiv.org/abs/2607.02840

[161] T. Daniel, C. Qi, D. Haramati, A. Zadeh, C. Li, A. Tamar, D. Pathak, and D. Held, "Latent particle world models: Self-supervised object-centric stochastic dynamics modeling," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2026, oral presentation. arXiv:2603.04553. [Online]. Available: https://openreview.net/forum?id=lTaPtGiUUc

[162] B. Qiu, L. Chen, Y. Liao, N. Wang, L. Wang, J. Luo, W. Zhao, S. Chen, D. Chen, Y. Li, C. Gao, S. Yan, S. Liu, M. Yao, and G. Ren, "GE-Sim 2.0: A roadmap towards comprehensive closed-loop video world simulators for robotic manipulation," *arXiv preprint arXiv:2605.27491*, 2026, arXiv:2605.27491 [cs.RO]. [Online]. Available: https://arxiv.org/abs/2605.27491

[163] S. Ye, J. Jang, B. Jeon, S. Joo, J. Yang, B. Peng, A. Mandlekar, R. Tan, Y.-W. Chao, B. Y. Lin, L. Liden, K. Lee, J. Gao, L. Zettlemoyer, D. Fox, and M. Seo, "Latent action pretraining from videos," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2025, arXiv:2410.11758. [Online]. Available: https://arxiv.org/abs/2410.11758

[164] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," in *Proceedings of Robotics: Science and Systems (RSS)*, Daegu, Republic of Korea, Jul. 2023, extended journal version: *Int. J. Robotics Research*, vol. 44, no. 10–11, pp. 1684–1704, 2025, doi:10.1177/02783649241273668. [Online]. Available: https://arxiv.org/abs/2303.04137

[165] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. P. Foster, P. R. Sanketi, Q. Vuong, T. Kollar, B. Burchfiel, R. Tedrake, D. Sadigh, S. Levine, P. Liang, and C. Finn, "OpenVLA: An open-source vision-language-action model," in *Proc. 8th Conf. Robot Learning (CoRL)*, ser. Proceedings of Machine Learning Research, vol. 270. Munich, Germany: PMLR, 2024, pp. 2679–2713, arXiv:2406.09246.

[166] A. O'Neill *et al.*, "Open X-Embodiment: Robotic learning datasets and RT-X models," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2024, pp. 6892–6903, open X-Embodiment Collaboration; Best Conference Paper Award; arXiv:2310.08864.

[167] T. Yuan, Z. Dong, Y. Liu, and H. Zhao, "Fast-WAM: Do world action models need test-time future imagination?" *arXiv preprint arXiv:2603.16666*, Mar. 2026, project page: https://yuantianyuan01.github.io/FastWAM/. [Online]. Available: https://arxiv.org/abs/2603.16666

[168] A. Hu, G. Corrado, N. Griffiths, Z. Murez, C. Gurau, H. Yeo, A. Kendall, R. Cipolla, and J. Shotton, "Model-based imitation learning for urban driving (MILE)," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2022, arXiv:2210.07729.

[169] L. Russell, A. Hu, L. Bertoni, G. Fedoseev, J. Shotton, E. Arani, and G. Corrado, "GAIA-2: A controllable multi-view generative world model for autonomous driving," *arXiv preprint arXiv:2503.20523*, 2025.

[170] X. Wang, Z. Zhu, G. Huang, X. Chen, J. Zhu, and J. Lu, "Drive-Dreamer: Towards real-world-driven world models for autonomous

driving," in *European Conference on Computer Vision (ECCV)*, 2024, arXiv:2309.09777.

[171] G. Zhao, X. Wang, Z. Zhu, X. Chen, G. Huang, X. Bao, and X. Wang, "DriveDreamer-2: Llm-enhanced world models for diverse driving video generation," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 39, no. 10, 2025, pp. 10 412–10 420, arXiv:2403.06845.

[172] F. Jia, W. Mao, Y. Liu, Y. Zhao, Y. Wen, C. Zhang, X. Zhang, and T. Wang, "ADriver-I: A general world model for autonomous driving," *arXiv preprint arXiv:2311.13549*, 2023.

[173] Y. Wang, J. He, L. Fan, H. Li, Y. Chen, and Z. Zhang, "Driving into the future: Multiview visual forecasting and planning with world model for autonomous driving (Drive-WM)," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, arXiv:2311.17918.

[174] J. Yang, S. Gao, Y. Qiu, L. Chen, T. Li, B. Dai, K. Chitta, P. Wu, J. Zeng, P. Luo, J. Zhang, A. Geiger, Y. Qiao, and H. Li, "Generalized predictive model for autonomous driving (GenAD)," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, arXiv:2403.09630.

[175] S. Gao, J. Yang, L. Chen, K. Chitta, Y. Qiu, A. Geiger, J. Zhang, and H. Li, "Vista: A generalizable driving world model with high fidelity and versatile controllability," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2024, arXiv:2405.17398.

[176] Wayve, "GAIA-3: Scaling world models to power safety and evaluation," Wayve Technical Blog, Dec. 2025, accessed: 2026-08-12. [Online]. Available: https://wayve.ai/thinking/gaia-3/

[177] D. Chen, V. Koltun, and P. Krähenbühl, "Learning to drive from a world on rails," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2021, arXiv:2105.00636.

[178] Wayve, "GAIA-4: Multimodal world models powering closed-loop simulation for safe and scalable autonomy," Wayve Technical Blog, Aug. 2026, published 3 Aug. 2026. [Online]. Accessed: 12 Aug. 2026. [Online]. Available: https://wayve.ai/thinking/gaia-4/

[179] C. M. Jiang, X. Masotto, and B. Sun, "The Waymo World Model: A New Frontier for Autonomous Driving Simulation," Waymo Blog, Feb. 2026, accessed: 2026-08-12. [Online]. Available: https://waymo.com/blog/2026/02/the-waymo-world-model-a-new-frontier-for-autonomous-driving-simulation/

[180] A. Basant, A. Kar, D. Paschalidou, F. Wei, F. Ferroni *et al.*, "NVIDIA OmniDreams: Real-time generative world model for closed-loop autonomous vehicle simulation," *arXiv preprint arXiv:2606.03159*, 2026, nVIDIA technical report; publicly released as NVIDIA Cosmos-Dreams (code: https://github.com/nv-tlabs/omni-dreams, weights: https://huggingface.co/nvidia/omni-dreams-models). [Online]. Available: https://arxiv.org/abs/2606.03159

[181] W. Zheng, W. Chen, Y. Huang, B. Zhang, Y. Duan, and J. Lu, "OccWorld: Learning a 3d occupancy world model for autonomous driving," in *European Conference on Computer Vision (ECCV)*, 2024, arXiv:2311.16038.

[182] L. Zhang, Y. Xiong, Z. Yang, S. Casas, R. Hu, and R. Urtasun, "Copilot4D: Learning unsupervised world models for autonomous driving via discrete diffusion," in *International Conference on Learning Representations (ICLR)*, 2024, arXiv:2311.01017.

[183] J. Lu, Z. Huang, J. Zhang, Z. Yang, and L. Zhang, "WoVoGen: World volume-aware diffusion for controllable multi-camera driving scene generation," in *European Conference on Computer Vision (ECCV)*, 2024, arXiv:2312.02934.

[184] C. Min, D. Zhao, L. Xiao, J. Zhao, X. Xu, Z. Zhu, L. Jin, J. Li, Y. Guo, J. Xing, L. Jing, Y. Nie, and B. Dai, "DriveWorld: 4d pre-trained scene understanding via world models for autonomous driving," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, arXiv:2405.04390.

[185] Z. Yang, Y. Chen, J. Wang, S. Manivasagam, W.-C. Ma, A. J. Yang, and R. Urtasun, "UniSim: A neural closed-loop sensor simulator," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, arXiv:2308.01898.

[186] A. Elluswamy and Tesla AI, "A unified world simulator for full self-driving and Optimus," Tesla (public disclosure), 2025, https://www.tesla.com/AI.

[187] J. Oh, X. Guo, H. Lee, R. Lewis, and S. Singh, "Action-conditional video prediction using deep networks in Atari games," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2015.

[188] S. Chiappa, S. Racaniere, D. Wierstra, and S. Mohamed, "Recurrent environment simulators," in *International Conference on Learning Representations (ICLR)*, 2017.

[189] SIMA Team, Google DeepMind, "Scaling instructable agents across many simulated worlds (SIMA)," Google DeepMind, 2024, arXiv:2404.10179.

[190] Y. Zhang *et al.*, "Matrix-Game: Interactive world foundation model," *arXiv preprint arXiv:2506.18701*, 2025.

[191] J. Yu, Y. Qin, X. Wang, P. Wan, D. Zhang, and X. Liu, "GameFactory: Creating new games with generative interactive videos," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025, arXiv:2501.08325.

[192] J. Li, J. Tang, Z. Xu *et al.*, "Hunyuan-GameCraft: High-dynamic interactive game video generation with hybrid history condition," *arXiv preprint arXiv:2506.17201*, 2025.

[193] M. Yang *et al.*, "Playable game generation," *arXiv preprint arXiv:2412.00887*, 2024.

[194] Virtuals Protocol GAME Team, "Video game generation: A practical study using Mario (MarioVGG)," Virtuals Protocol technical report, 2024, https://virtual-protocol.github.io/mario-videogamegen/.

[195] J. Tang, J. Liu, J. Li, L. Wu, H. Yang, P. Zhao, S. Gong, X. Yuan, S. Shao, L. Zhang, and Q. Lu, "Hunyuan-gamecraft-2: Instruction-following interactive game world model," arXiv preprint arXiv:2511.23429, 2025, tencent Hunyuan technical report; revised Feb. 2026. [Online]. Available: https://arxiv.org/abs/2511.23429.

[196] D. Rivas, E. Breece, and S. Chambers, "Project Genie: Experimenting with infinite, interactive worlds," Google Blog, Google Labs and Google DeepMind, Jan. 2026, published 29 January 2026. Interactive world-exploration prototype powered by the Genie 3 world model; available to Google AI Ultra subscribers in the U.S. (18+), with generations limited to 60 seconds. Accessed 12 August 2026. https://blog.google/innovation-and-ai/models-and-research/google-deepmind/project-genie/.

[197] D. Rivas, J. Herbert, and N. Segaran, "Simulate real-world places with Project Genie and Street View," Google Blog, Google Labs, Google DeepMind and Google Maps, May 2026, published 19 May 2026. Grounds Project Genie world generation in "nearly 20 years of Google Street View imagery" through Maps Imagery Grounding; available for places in the U.S., with the prototype rolling out to Google AI Ultra subscribers globally (18+). Accessed 12 August 2026. https://blog.google/innovation-and-ai/models-and-research/google-deepmind/project-genie-expands/.

[198] A. Bolton, A. Lerchner, A. Cordell, A. Lampinen, C. Lu, J. Clune, V. Mnih, R. Hadsell, D. Hassabis, Z. Ghahramani *et al.*, "SIMA 2: A generalist embodied agent for virtual worlds," *arXiv preprint arXiv:2512.04797*, 2025, SIMA Team, Google DeepMind; authors listed alphabetically. [Online]. Available: https://arxiv.org/abs/2512.04797

[199] Microsoft Research, "WHAMM: Real-time world modelling of interactive environments," Microsoft Research Article, 2025, https://www.microsoft.com/en-us/research/articles/whamm-real-time-world-modelling-of-interactive-environments/.

[200] F.-F. Li, "From words to worlds: Spatial intelligence is AI's next frontier," World Labs, 2025, https://www.worldlabs.ai/blog.

[201] J. Xiang *et al.*, "Pandora: Towards general world model with natural language actions and video states," *arXiv preprint arXiv:2406.09455*, 2024.

[202] H. Tang, D. Key, and K. Ellis, "WorldCoder, a model-based LLM agent: Building world models by writing code and interacting with the environment," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2024, arXiv:2402.12275.

[203] FAIR CodeGen Team, Meta, "CWM: An open-weights LLM for research on code generation with world models," *arXiv preprint arXiv:2510.02387*, 2025.

[204] H. Chae *et al.*, "Web agents with world models: Learning and leveraging environment dynamics in web navigation," in *International Conference on Learning Representations (ICLR)*, 2025, arXiv:2410.13232.

[205] Y. Zuo, Z. Xiao, L. Sheng, F. Huang, J. Tu, Y. Liu, T. Tang, X. Hu *et al.*, "Qwen-AgentWorld: Language world models for general agents," *arXiv preprint arXiv:2606.24597*, 2026, qwen Team, Alibaba Group. [Online]. Available: https://arxiv.org/abs/2606.24597

[206] M. G. Bellemare, Y. Naddaf, J. Veness, and M. Bowling, "The arcade learning environment: An evaluation platform for general agents," *Journal of Artificial Intelligence Research (JAIR)*, vol. 47, pp. 253–279, 2013.

[207] Y. Tassa, Y. Doron, A. Muldal, T. Erez, Y. Li, D. de Las Casas, D. Budden, A. Abdolmaleki, J. Merel, A. Lefrancq, T. Lillicrap, and M. Riedmiller, "DeepMind control suite," *arXiv preprint arXiv:1801.00690*, 2018.

[208] E. Todorov, T. Erez, and Y. Tassa, "MuJoCo: A physics engine for model-based control," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012.

[209] D. Hafner, "Benchmarking the spectrum of agent capabilities (Crafter)," in *International Conference on Learning Representations (ICLR)*, 2022, arXiv:2109.06780.

[210] W. H. Guss, B. Houghton, N. Topin, P. Wang, C. Codel, M. Veloso, and R. Salakhutdinov, "MineRL: A large-scale dataset of Minecraft demonstrations," in *International Joint Conference on Artificial Intelligence (IJCAI)*, 2019, arXiv:1907.13440.

[211] L. Fan, G. Wang, Y. Jiang, A. Mandlekar, Y. Yang, H. Zhu, A. Tang, D.-A. Huang, Y. Zhu, and A. Anandkumar, "MineDojo: Building open-ended embodied agents with internet-scale knowledge," in *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks*, 2022, arXiv:2206.08853, Outstanding Paper Award.

[212] S. James, Z. Ma, D. R. Arrojo, and A. J. Davison, "RLBench: The robot learning benchmark & learning environment," *IEEE Robotics and Automation Letters (RA-L)*, 2020, arXiv:1909.12271.

[213] T. Yu, D. Quillen, Z. He, R. Julian, K. Hausman, C. Finn, and S. Levine, "Meta-World: A benchmark and evaluation for multi-task and meta reinforcement learning," in *Conference on Robot Learning (CoRL)*, 2019, arXiv:1910.10897.

[214] K. Cobbe, C. Hesse, J. Hilton, and J. Schulman, "Leveraging procedural generation to benchmark reinforcement learning (Procgen)," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2020, arXiv:1912.01588.

[215] R. Agarwal, M. Schwarzer, P. S. Castro, A. Courville, and M. G. Bellemare, "Deep reinforcement learning at the edge of the statistical precipice," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021, arXiv:2108.13264, Outstanding Paper Award.

[216] T. Unterthiner, S. van Steenkiste, K. Kurach, R. Marinier, M. Michalski, and S. Gelly, "Towards accurate generative models of video: A new metric & challenges (FVD)," *arXiv preprint arXiv:1812.01717*, 2018.

[217] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity (SSIM)," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[218] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric (LPIPS)," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, arXiv:1801.03924.

[219] J. Hessel, A. Holtzman, M. Forbes, R. Le Bras, and Y. Choi, "CLIPScore: A reference-free evaluation metric for image captioning," in *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2021, arXiv:2104.08718.

[220] D. M. Bear, E. Wang, D. Mrowca, F. J. Binder, H.-Y. F. Tung, R. T. Pramod, C. Holdaway, S. Tao, K. Smith, F.-Y. Sun, L. Fei-Fei, N. Kanwisher, J. B. Tenenbaum, D. L. K. Yamins, and J. E. Fan, "Physion: Evaluating physical prediction from vision in humans and machines," in *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks*, 2021, arXiv:2106.08261.

[221] H.-Y. Tung, M. Ding, Z. Chen, D. Bear, C. Gan, J. B. Tenenbaum, D. L. K. Yamins, J. E. Fan, and K. A. Smith, "Physion++: Evaluating physical scene understanding that requires online inference of different physical properties," in *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks*, 2023, arXiv:2306.15668.

[222] H. Duan, H.-X. Yu, S. Chen, L. Fei-Fei, and J. Wu, "WorldScore: A unified evaluation benchmark for world generation," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025, arXiv:2504.00983.

[223] Z. Huang, Y. He, J. Yu, F. Zhang, C. Si, Y. Jiang, Y. Zhang, T. Wu, Q. Jin, N. Chanpaisit, Y. Wang, X. Chen, L. Wang, D. Lin, Y. Qiao, and Z. Liu, "VBench: Comprehensive benchmark suite for video generative models," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, arXiv:2311.17982.

[224] D. Zheng, Z. Huang, H. Liu, K. Zou, Y. He, F. Zhang, L. Gu, Y. Zhang, J. He, W.-S. Zheng, Y. Qiao, and Z. Liu, "VBench-2.0: Advancing video generation benchmark suite for intrinsic faithfulness," *arXiv preprint arXiv:2503.21755*, 2025.

[225] Y. Li, Y. Zhu, J. Wen, C. Shen, and Y. Xu, "WorldEval: World model as real-world robot policies evaluator," *arXiv preprint arXiv:2505.19017*, 2025.

[226] J. Yuan, F. Pizzati, F. Pinto, L. Kunze, I. Laptev, P. Newman, P. Torr, and D. D. Martini, "LikePhys: Evaluating intuitive physics understanding in video diffusion models via likelihood preference," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2026, arXiv:2510.11512. [Online]. Available: https://arxiv.org/abs/2510.11512

[227] Y. Yu, S. Zhang, Y. Sheng, H. Ren, and H. Lin, "How should world models be evaluated for embodied decision-making? A decision-making-centric position," *arXiv preprint arXiv:2606.15032*, 2026. [Online]. Available: https://arxiv.org/abs/2606.15032

[228] E. Karcini, F. Mehrban, Q. Nguyen, M. Schwager, A. Ajoudani, C. Cadena, J. Peters, M. Hutter, and H. Bou-Ammar, "Robots need more than VLA and world models," *arXiv preprint arXiv:2606.06556*, 2026.

[229] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn *et al.*, "π_0: A vision-language-action flow model for general robot control," *arXiv preprint arXiv:2410.24164*, 2024.

[230] Z. Zhang, Z. Li, B. Rahmati, R. H. Yang, Y. Ma, A. Rasouli, S. Pakdamansavoji, Y. Wu, L. Zhang, T. Cao, F. Wen, X. Wang, X. Quan, and Y. Zhang, "Do world action models generalize better than VLAs? a robustness study," *arXiv preprint arXiv:2603.22078*, 2026.

[231] N. Hansen and X. Wang, "Hallucination in world models is predictable and preventable," *arXiv preprint arXiv:2606.27326*, Jun. 2026. [Online]. Available: https://arxiv.org/abs/2606.27326