

LARGE VIDEO PLANNER ENABLES GENERALIZABLE ROBOT CONTROL

Boyuan Chen^{1*} Tianyuan Zhang^{1*} Haoran Geng^{2*} Kiwhan Song¹ Caiyi Zhang²
 Peihao Li² William T. Freeman¹ Jitendra Malik² Pieter Abbeel² Russ Tedrake¹
 Vincent Sitzmann¹ Yilun Du³
¹MIT ²UC Berkeley ³Harvard

ABSTRACT

General-purpose robots require decision-making models that generalize across diverse tasks and environments. Recent works build robot foundation models by extending multimodal large language models (MLLMs) with action outputs—creating vision–language–action (VLA) systems. These efforts are motivated by the intuition that MLLMs’ large-scale language and image pretraining can be effectively transferred to the action output modality. In this work, we explore an alternative paradigm of using large-scale video pretraining as a primary modality for building robot foundation models. Unlike static images and language, videos capture spatio-temporal sequences of states and actions in the physical world that are naturally aligned with robotic behavior. We curate an internet-scale video dataset of human activities and task demonstrations, and train, for the first time at a foundation-model scale, an open video model for generative robotics planning. The model produces zero-shot video plans for novel scenes and tasks, which we post-process to extract executable robot actions. We evaluate task-level generalization through third-party selected tasks in the wild and real-robot experiments, demonstrating successful physical execution. Together, these results show robust instruction following, strong generalization, and real-world feasibility. We release both the model and dataset to support open, reproducible video-based robot learning. Our website is available at <https://www.boyuan.space/large-video-planner/>.

1 INTRODUCTION

A key component of many robotic systems is the planning algorithm [30], which takes a high-level task or instruction alongside the robot’s sensory observations to generates a sequence of states and actions that will achieve the goal. General-purpose robots — systems designed to operate reliably across diverse tasks and novel environments — would greatly benefit from planning algorithms that are themselves extremely general. Such planning algorithms should be able to comprehend unseen tasks, adapt fluidly to novel scenes, and output physically coherent behaviors. Developing these strong generalization capabilities remains a central, unresolved challenge of embodied intelligence today.

The recent success of foundation models in language and vision has reshaped how generalization is achieved in AI. Large language models (LLMs) [1, 5, 79, 21, 53, 77] trained on internet-scale text corpora exhibit broad competence across unseen tasks, suggesting that scale and data diversity can induce powerful transfer. Extending this idea, multimodal large language models (MLLMs) [54, 75, 6, 35] align vision and language, grounding textual reasoning in perception. These advances have inspired robot foundation models—large, unified architectures that generalize across scenes and tasks by integrating perception, language, and control. A key instantiation is the Vision–Language–Action (VLA) model [12, 43, 10], which extends MLLMs with an action output modality.

However, in comparison to web-scale text and image data that underpin MLLMs [70, 64, 18], robot action data is much scarcer [20, 15]. As a result, it is difficult to build VLA models with the same level of competency as that of MLLMs, with existing VLAs relying on an asymmetric form of transfer, where the pretrained knowledge in an MLLM is finetuned on a narrow amount of robot data. Such a construction leads to poor generalization when given new robot tasks in unseen situations [83].

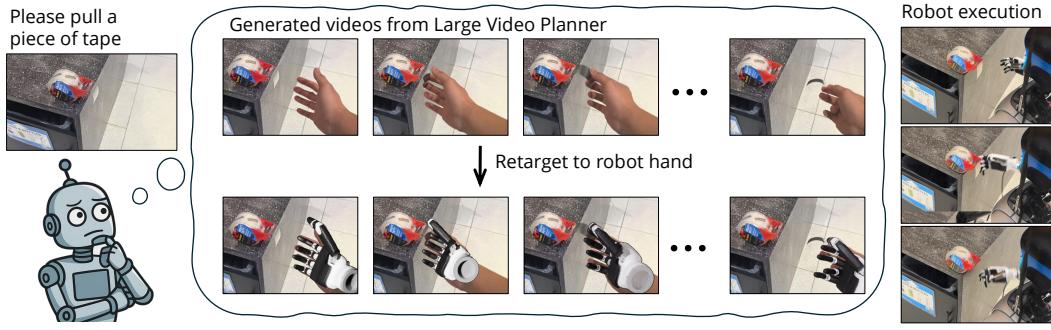


Figure 1: Autonomous Robot Execution with Large Video Planner. Our approach uses video generation as a visual motion planner in pixel space. From a single image and a task instruction, the model generates a video depicting how the task should be completed. The predicted human motion is then retargeted to a robot hand for real-world execution, enabling zero-shot visual planning in diverse scenes.

In this work, we propose an alternative paradigm for robot foundation models—using video as the primary modality. Unlike static image–text pairs, videos naturally encode state–action plans, visually depicting how the world evolves as agents interact with it. A video generative model conditioned on a textual instruction and an initial observation frame can predict plausible future frames—effectively generating visual action plans for diverse tasks. This formulation aligns closely with robotics: it captures spatial and temporal continuity, offering a far richer representation of continuous actions than text tokens. Moreover, video data is abundant online, spanning human activities, instructional tutorials, and task demonstrations. Each video implicitly contains action information, following the same pretraining principle that powers MLLMs—leveraging large, naturally occurring data to learn mappings grounded in real-world behavior. Compared to the asymmetric transfer of VLAs, this video-based paradigm offers a directly grounded source of transfer: the data itself captures the temporal dynamics of action, providing a stronger bridge to downstream embodied tasks.

To instantiate this paradigm, we develop a video foundation model purpose-built for embodied decision-making and action extraction. Unlike existing video generation models—typically optimized for content creation and prone to mode collapse—we prioritize the emphasis on physical consistency to real image observations and adherence to task instructions. It takes textual instructions and initial observation frame(s) as input, generating predictive video plans from which executable actions are extracted [26].

We realize this vision through advances in both data and model design. First, we introduce and release a large-scale, open dataset curated for embodied decision making. Sourced from a diverse mix of internet and robot videos, this dataset is carefully processed to capture complex human and robot behaviors, maintain high temporal coherence, and ensure tight alignment with language instructions. Second, we propose a novel model that leverages History Guidance [73] and Diffusion Forcing [16] to specifically enhance temporal coherence and causal consistency in the generated frames. These data and model contributions jointly enable large-scale, temporally grounded pretraining, leading to robust generalization across diverse tasks and scenes.

We evaluate our model’s generalization through two complementary experiments. First, we perform an extensive independent evaluation to assess task-level generalization of the video model itself: independent testers freely selected scenes and tasks—encouraged to be creative and challenging—producing evaluations spanning diverse conditions, from outdoor scenes like crosswalks to dexterous tasks such as tearing tape. Second, we conduct real-robot experiments demonstrating end-to-end execution. Actions extracted from generated video plans are deployed on physical robots, from parallel grippers to dexterous hands, successfully executing tasks in uncontrolled environments. Compared to baseline models, our approach exhibits a stronger grasp of contact dynamics, goal reasoning, and end-to-end execution, demonstrating robust generalization across both simulated and real settings.

In summary, our work makes three primary contributions. (1) Large Video Planner (LVP), a large-scale video foundation model designed for robotic manipulation, and an associated framework for deploying it as a zero-shot policy on real robots. (2) A curated, open internet-scale video dataset of human activities and robot task demonstrations, carefully processed for embodied decision making and instruction following. (3) A rigorous evaluation of task-level generalization, using an independent

testing protocol and real-robot experiments to systematically assess generalization across unseen environments, tasks, and embodiments.

2 RELATED WORK

Video Diffusion. Diffusion models [72, 37, 52] currently represent the state of the art for synthesizing high-quality videos [38, 11, 14, 81]. They generate videos by iteratively denoising an entire sequence of frames jointly, often operating in a lower-dimensional latent space for efficiency. Recent works adapt this process to better capture the temporal structure, yielding improved results over vanilla full-sequence diffusion [68, 16, 87, 41]. Rolling Diffusion [68] introduces a sliding-window diffusion process that progressively corrupts data from past to future, enabling autoregressive rollouts for long video generation. Diffusion Forcing [16] generalizes this idea by training with independently randomized noise levels across tokens, allowing the model to generate sequences in arbitrary order and with next-token or next-few-token denoising.

To improve visual conditional generation, classifier-free guidance [36] has proven to be an essential technique to improve fidelity and controllability. It is mostly applied with text based conditional signals. History Guidance [73] extends this paradigm by conditioning generation on one or several preceding frames, amplifying temporal grounding. This approach has been applied in tasks such as image-to-video generation and video extension [88, 73], where temporal grounding is critical.

Robot Foundation Models. A large body of recent work has focused on constructing robot foundation models from large-scale robotics datasets [20, 42, 27, 80, 46, 32]. Such models include *Vision–Language–Action (VLA) policies* [13, 78, 43, 61, 9, 76, 76], which directly map multimodal inputs to robot actions, as well as *large diffusion and transformer-based networks* that learn to generate trajectories or action sequences [91, 7]. Another line of work [3, 31, 23, 48, 40] explores *embodied LLMs and planner–policy hybrids*, which leverage large language models for high-level reasoning and planning while delegating execution to low-level controllers, e.g., PaLM-E [24] and RoboFlamingo [49].

In contrast to these approaches, we formulate a robot foundation model as a *video generation model* that produces realistic interaction videos depicting what the robot should do, from which executable actions can then be extracted and retargeted.

Video Generation for Robotics. Learning from video demonstration has been an interesting research direction for robot learning [55, 84, 71, 59], partially due to its richness of data. With recent advances in video generation, researchers have begun exploring its use in robotics. One line of work employs video generation for visual policies, where models synthesize videos of successful task completion to guide control [26, 51, 8, 56, 2]. Another line treats video generation as a dynamics model, predicting future frames conditioned on action inputs [4, 85, 25, 90, 65]. Video world models have also been used as evaluators, with the potential of enabling robots to validate and refine their strategies more efficiently through simulated rollouts [60]. The distinction between policy learning and dynamics modeling is increasingly blurred, as recent approaches train unified models that jointly generate both videos and robot actions [47].

Given a generated video, there are several approaches to extract continuous actions. One approach is to learn an inverse dynamics or goal-conditioned policy on top of the generated videos [26, 82, 39, 57, 89]. Alternatively, we can directly infer 2D or 3D scene flow to obtain actions [44, 17]. In contrast, in this paper we illustrate how we can leverage 3D scene reconstruction MegaSam [50] followed by hand reconstruction using HaMeR [62] and Dex-Retargeting [66] to obtain actions.

3 METHOD

A robot foundation model maps observations and goals to a sequence of actions. We realize this through a two-stage design: a large video planner followed by action extraction. Consider a robot facing a door it has never encountered before. Its camera perceives the door handle as its owner instructs, “Open this door.” The robot first employs a video foundation model to imagine how a rational human would perform the task—generating a video where a hand reaches for the handle, twists it, and pushes the door open. It then applies action extraction algorithms to translate this visual plan into executable control signals, whether for a dexterous five-fingered hand or a parallel gripper.

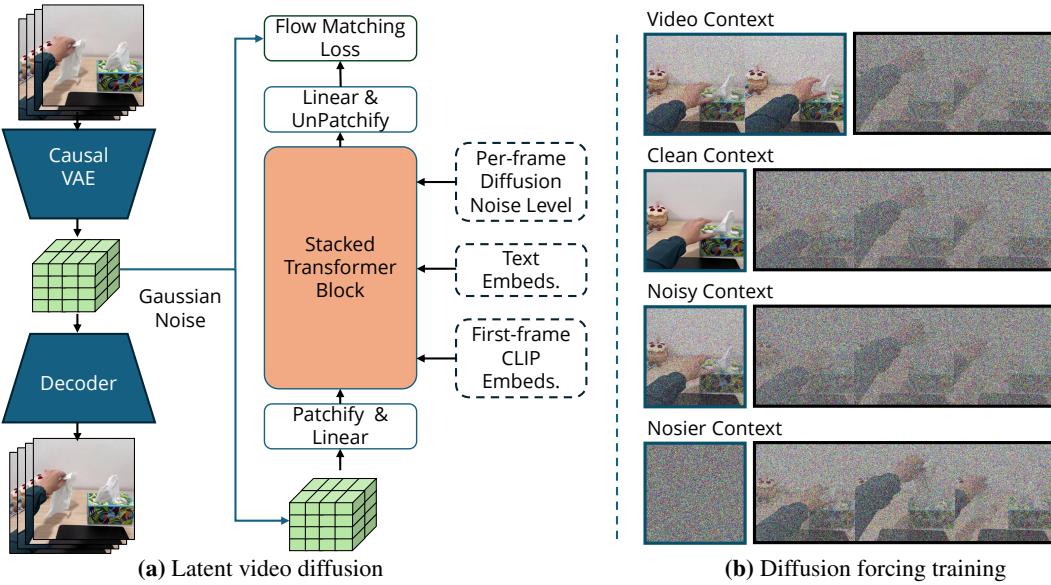


Figure 2: **LVP Overview:** (a) Overview of the latent video diffusion framework. We first use a temporally causal VAE to encode video clips into compressed 3D latent representations. Then we train a diffusion transformer in this latent space with flow matching objectives. (b) We jointly train image-to-video (I2V) and video-to-video (V2V) with a modified diffusion forcing training strategy. During training, a random context length between 0 and 6 frames is selected, dividing the video into history and future segments. Two independent noise levels are applied to these segments, and the history segment is set to zero noise with a 50% probability. We visualize four representative cases of this noisy training strategy: the top row shows that longer contexts enable V2V training; the second row shows clean first-frame contexts, which exactly aligns with standard I2V training; and the bottom two rows show noisy context frames, which improve robustness to out-of-distribution conditioning.

In Section 3.1, we present our video foundation model for generative planning in video space. Section 3.2 introduces an internet-scale video dataset of human activities and robot demonstrations that we curated for this study. Finally, Section 3.3 describes our retargeting mechanism that grounds the generated video plans into robot-specific actions across diverse morphologies.

3.1 LVP: A VIDEO FOUNDATION MODEL FOR GENERATIVE PLANNING

Latent Diffusion. We begin building our video foundation model following the latent diffusion framework [14, 81]. We use a temporally causal 3D variational autoencoder (VAE) to compress a video clip in pixel space into a compact, lower-dimensional latent representation x . The VAE encodes each $8 \times 8 \times 4$ spatiotemporal patch into a 16-channel embedding, converting an input of shape $[1 + T, 3, H, W]$ into a latent of shape $[1 + \lceil T/4 \rceil, 16, \lceil H/4 \rceil, \lceil W/4 \rceil]$, where $T + 1$ is the number of frames and H, W are spatial dimensions. The first frame of a video is repeated 4 times before such compression to allow co-training with single-frame image data, which corresponds to the 1 in $T + 1$.

We then freeze the 3D VAE and train a special video diffusion model [72, 37, 52, 63] in this compressed latent space using a modified Diffusion Forcing Transformer [16, 73], a DiT [63] variant we introduce below, and illustrated in Figure 2. Following the diffusion training recipe, we add Gaussian noise to a clean video latent and train our diffusion model to remove such noise. At sampling time, starting from a latent pre-filled with noise, the model iteratively denoises the latent until obtaining a clean sample. The VAE decoder then decodes this latent into a video sample.

Specifically, we train this video diffusion model with the flow matching objective [52]. Under a shifted schedule [28] that emphasizes higher noise levels, we add noise to an encoded video latent z_0 by $z_k = (1 - k)z_0 + k\epsilon$, where k denotes the chosen noise level, $\epsilon \sim \mathcal{N}(0, 1)$ and z_k is the noisy latent. The model f_θ is trained to predict the flow $\epsilon - z_0$ conditioned on the noisy latent z_t , conditioning c (comprising the input image and text instruction), and noise level t , minimizing the matching loss [52] $\mathcal{L} = \|f_\theta(z_k, c, k) - k(\epsilon - z_0)\|_2$.

Diffusion Forcing Transformer. A challenge in the video diffusion model is temporal coherence. In our formulation, the generated video must be coherent with not only the language instruction

(text-to-video or t2v) but also the first frame (image-to-video or i2v) specified by the robot observation. In addition, one may want to condition the video generation on multiple previous frames (video-to-video or v2v) to generate multi-stage video plans. Traditionally, one achieves such conditioning by finetuning a t2v model to cross-attend to separate patches of the context frame(s) [81]. We, however, propose to better satisfy this need with the recently proposed Diffusion Forcing framework.

Instead of adding a uniform level of noise to all tokens like in legacy video diffusion models, Diffusion Forcing [16] found that training video diffusion models with different noise levels at different frames has the additional benefit of flexibility and rollout stability. Since all the noise levels are random during training, at test time one can flexibly control the conditioning by selecting the desired noise level.

To learn i2v and v2v with a unified objective, we adopt diffusion forcing and apply different noise levels to context frames versus generated frames. As shown in Figure 2(b), given a diffusion transformer on a fixed number of latent frames, we first randomly sample a history length from $\{0, 1, 2, \dots, 6\}$ latent frames, splitting the video into a history segment and a future segment. We then apply independent noise levels to each segment and feed the resulting noisy video to our model, leaving all other settings unchanged. For example, if one adds zero noise to the first frame or first few frames at training time, the model will find it as a perfectly visible context frame and learn to condition on it; if the history frames have an intermediate noise level, the model treats it as partial information and learns to be robust to out-of-distribution context frames. In this way, we can flexibly condition on a clean first frame or multiple history frames at sampling time, by setting their noise levels to 0.

Not only does this method eliminate an extra cross-attention to variable-length context tokens, but it's also compatible with existing DiT model weights without architectural changes. Following Song et al. [73], we simply feed different noise level embeddings to different tokens in the DiT architecture, instead of uniform ones. This allows us to train a Diffusion Forcing model on top of the weights of a pre-trained video foundation model, WAN 2.1 14B [81]. Following the practice of WAN 2.1 14B, we cross-attend to the CLIP features of the first frame as well as the text embeddings extracted by the UMT5 [19] encoder. Because Diffusion Forcing achieves context frame conditioning in a cleaner way, we remove WAN's mask and guidance channels used for image conditioning.

Enhanced Temporal Coherence with History Guidance. In addition to flexible conditioning and compatibility with legacy weights, our design can significantly enhance context coherence by enabling special sampling techniques from Diffusion Forcing [16, 73].

Classifier-Free Guidance (CFG) [36] is known to improve visual quality and conditioning adherence in visual generative models. WAN 2.1 utilizes a text-CFG that combines the output of a text-conditional diffusion model and that of an unconditional one. However, this still yields unsatisfactory motion fidelity and weak image conditioning as shown in Figure 5.

LVP adopts *history guidance* [73], a CFG variant that performs guidance on any amount of context frames. Let x_k denote the future segment to be diffused at noise level k and c_{text} the task instruction. As our model is trained with Diffusion Forcing, we can flexibly condition on a provided history segment x_{hist} at sampling time by setting its noise level to zero, be it a single frame or a context video:

$$\nabla \log p(x_k | x_{\text{hist}}, c_{\text{text}}, k). \quad (1)$$

Similarly, we can set the noise level of context frames to the maximum to fully mask out the context frames and obtain the unconditional score:

$$\nabla \log p(x_k | c_{\text{text}}, k). \quad (2)$$

To perform history guidance, we sample with the combined score

$$s_{\text{hist}} = (1 + w_{\text{hist}}) \nabla \log p(x_k | x_{\text{hist}}, c_{\text{text}}) - w_{\text{hist}} \nabla \log p(x_k | c_{\text{text}}). \quad (3)$$

Just as text-based CFG enhances adherence to text instruction, history guidance enhances adherence to context images. During sampling, we combine both history guidance and text-based CFG to generate videos that adhere to both text and context frames by sampling with the score

$$s_{\text{final}} = (1 + w_{\text{hist}}) \nabla \log p(x_k | x_{\text{hist}}, c_{\text{text}}) - w_{\text{hist}} \nabla \log p(x_k | c_{\text{text}}) + \\ (1 + w_{\text{text}}) \nabla \log p(x_k | x_{\text{hist}}, c_{\text{text}}) - w_{\text{text}} \nabla \log p(x_k | x_{\text{hist}}) \quad (4)$$

This combined guidance technique can significantly enhance the plan quality compared to traditional text-based guidance, yielding physically viable plans with strong instruction following.

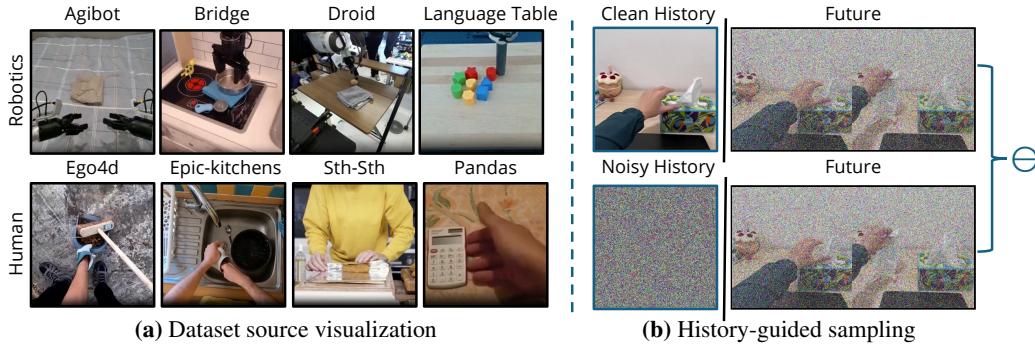


Figure 3: **(a)** Visualization of our eight dataset sources. First row: four robotics datasets. Second row: four human-centric datasets. **(b)** Illustration of our video diffusion sampling strategy, where scores estimated with and without history are linearly combined. Text conditioning and the diffusion transformer are omitted for clarity.

Autoregressive Extension for Multi-Stage Planning. Due to the flexible history conditioning, our model can extend a previously generated or captured video. The model supports up to 24 frames (6 latent frames in VAE space) as context. We can repeat video extension iteratively to generate multi-stage video plans. See Figure 7 and the videos on our website for multi-stage results.

Training Details. We train the model in two stages to progressively improve its visual planning capability and visual quality:

- **Continue pretraining.** Starting from Wan I2V 14B weights, we discard the weights that handle the extra masking and image guidance channels. We train on the full dataset for 60k steps with a batch size of 128, for a total of 200B tokens. At this stage, the model captures rich dynamics and strong instruction-following behavior, but the generated videos often exhibit excessive camera motion, which hinders smooth deployments on robots.
- **Low camera motion finetuning.** To reduce unwanted camera motion, we curate a smaller subset from Ego4D, Epic-Kitchens, and Panda datasets by selecting clips with a much lower average optical flow magnitude, and finetune for an additional 10k steps. This stage effectively suppresses camera drift and improves overall temporal smoothness and visual stability.

The total training takes around 14 days with 128 H100 SXM5 GPUs.

3.2 LVP-1M : A VIDEO DATASET OF HUMAN AND ROBOT ACTIONS

Training a video foundation model for *embodied planning* demands abundant data emphasizing diverse object interactions with action-centric text annotations. This contrasts with standard video datasets used for content-creation-oriented video generation [45, 81], which often prioritize aesthetic quality, cinematic shots, or dense captions of visual appearance and elements rather than motions. To this end, we curate **LVP-1M**, a diverse and high-quality dataset of 1.4M short clips showing humans or robots interacting with objects, each paired with multiple action-centric captions.

Video Sources. Given the vast availability of video data, we source raw videos from existing datasets before providing our own high-quality annotations. To ensure broad diversity across scenes, tasks, and embodiments, we combine robot teleoperation and human activity videos.

We start with web crawls widely used by video foundation models. We choose Pandas 70M [18] as a source of videos for heavy filtering. These internet-scale datasets contain diverse videos filtered for visual quality and captioned with visual content. They provide crucial scale and diversity that span countless tasks, scenes, and objects. However, only a small proportion of these videos capture detailed hand interactions with objects at sufficient resolution, not to mention near-zero robot coverage.

A second source of video comes from egocentric human activity datasets. These medium-scale datasets contain many annotated human-object interactions with moderate diversity but often suffer from large background motion due to camera movement. In addition, we found that atomic action annotations in these datasets still have lengths varying from seconds to minutes. We opt to draw videos from Ego4D [34], Epic Kitchens [22], and Something-something [33] dataset before heavy filtering, frame alignment and recaptioning.

A third source of videos come from robotics datasets featuring teleoperated robots performing tasks. While they provide knowledge about robot morphology, spanning parallel-jaw grippers to dexterous multi-fingered hands, they often have poor visual quality, poorly aligned frame rates, and limited diversity. Further, we found that short captions describing the task are often lacking, with many videos annotated with captions as vague as “pick”, or not featuring a caption at all. We opt to select Bridge [80], Droid [42], Language Table [58], and AgiBot-World [15] for heavy captioning and frame alignment.

We provide a summary of video sources and their key properties in Table 1, examples from each dataset in Figure 3(a), and additional details in App. A. Together, we hope our model will achieve synergy by learning better instruction following from the diversity of web crawls, better object-hand interactions from egocentric human activities, as well as robotic morphologies from robot data.

Temporal Alignment. We train our model to generate 3-second action videos at 16 frames per second, as this provides a good trade-off between computational cost and action granularity. However, we observe that different datasets often feature varying-length clips for atomic actions, spanning 1 second to 1 minute. A closer examination reveals that robotics datasets contain motions much slower than those of humans performing the same tasks and are often recorded at drastically different frame rates, sometimes as low as 5 fps.

Rather than naively aligning frame rates or trimming a video clip to a target length, we deem it important to align all clips to human speed to avoid temporal inconsistency - if a human normally finishes the task in 3 seconds, we resample the robot video (via upsampling or speeding up) so that it performs the same task in 3 seconds, regardless of its original frame rate or teleoperation speed. We achieve this by visually inspecting all datasets to determine the appropriate subsampling ratio following this principle. We also break down long-horizon tasks into atomic actions if any annotation contains multi-stage tasks. Some egocentric human activity datasets already provide action clip annotations, but we further refine them by trimming each clip precisely at the action’s start and end points. As we found later in experiments, such alignment is critical to enhancing the transfer between different morphologies.

Quality Filtering. After temporal alignment, we first discard clips that are low-resolution, too short, too long, or poorly lit. We then apply some additional filters to focus the model on embodied motion planning:

Filtering rapid camera motions. Many egocentric videos exhibit rapid camera rotations, leading to large background shifts and high training loss. These distract the model from learning meaningful foreground object motions. To mitigate this, we filter videos using optical flow statistics.

Ensuring visible embodiment. To avoid ambiguity, we require the embodiment (hand or robot gripper) to be clearly visible in the first frame. We use object detectors to automatically filter out clips where the embodiment is absent.

Expert motion. Many robot datasets contain suboptimal trajectories where the robot does not successfully accomplish the task. Traditionally, robot foundation models do not filter such data even when a “success” annotation is provided. However, we consider it important to remove these failure trajectories.

Filtering Pandas-70M subset. We perform three stages of progressive filtering to extract a subset focused on human interactions from the large Panda-70M dataset. First, we perform keyword-based filtering on captions using a whitelist (e.g., “grasping”, “pull”) and a blacklist (e.g., “cartoon,” “video game”). We then use human detectors to retain only clips containing one to four humans visible in the first, middle, and last frames. Finally, we perform another round of filtering with Gemini. For each video clip, we prompt Gemini with four questions to verify whether the clip contains rich human hand motions. We list more details of this in the Appendix A.1.

Action-Centric Re-Caption. To enhance the instruction following of our model, we generate multiple high-quality captions for each video. Traditionally, video foundation models favor extremely detailed captions describing all the visual elements. We observe that some robot datasets or ego-centric datasets only feature extremely short task descriptions as simple as one word, such as “pick”. For these videos (e.g., DROID, Ego4D), we prompt Gemini Flash with the instruction and initial frame to create more detailed and varied captions. For videos that lack task annotations, we prompt Gemini with the entire video clip and ask it to describe the primary action and involved objects.

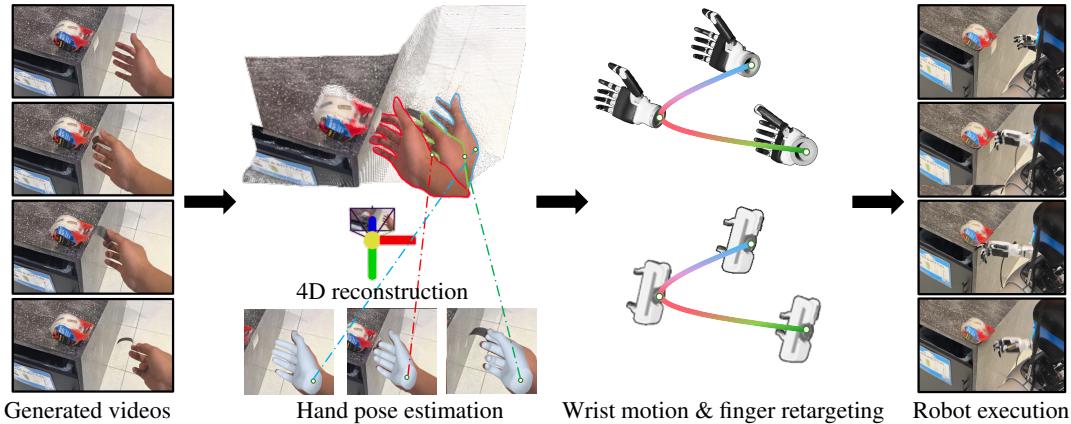


Figure 4: **Pipeline from Video to Action.** Given a generated video depicting a human hand performing a task, we first reconstruct and track the hand in 3D (second column). The reconstructed hand motion is then retargeted to dexterous hands or grippers (third column). Finally, the retargeted trajectory is transformed into the robot’s control frame and executed in the real world (rightmost column).

In total, including repeated captions, we obtain 4.1 million captioned clips. We find that providing Gemini with a short description (e.g., “pick up a cup”) substantially improves caption accuracy, as the video perception of Gemini tends to describe static scene contents well but struggles with fine-grained action dynamics. We make sure each video clip is paired with two to five distinct captions, some short, some extremely descriptive, to enhance linguistic diversity and improve training robustness.

Name	# Filtered clips	Robot?	Ego-centric?	Morphology	In-the-wild	Bimanual
Bridge	25k	Yes	Third-person	Gripper	No	No
DROID	192k	Yes	Third-person	Gripper	No	Yes
Language-Tables	71k	Yes	Third-person	Gripper	No	No
Agibot-World	863k	Yes	Third-person	Gripper	No	Yes
Ego4D	39k	No	Egocentric	Human Hand	Yes	Yes
Epic-Kitchens	7k	No	Egocentric	Human Hand	No	Yes
Something-Something	93k	No	Third-person	Human Hand	Yes	Yes
Panda-70M (filtered)	196k	No	Third-person	Human Hand	Yes	Yes

Table 1: **LVP-1M** sources and properties after curation. The dataset targets action-centric clips with broad diversity across embodiment (robot/human), viewpoint (ego/third-person), scene type (in-the-wild/lab), morphology, and bimanuality, totaling 1.4 million clips.

3.3 ROBOT ACTIONS FROM VIDEO PLANS

Given a camera observation and a task description, our large video planner can generate a video plan of a human hand or robot gripper executing the task. This section describes how we extract executable actions from a video plan and deploy them on robots.

Our action extraction pipeline supports retargeting generated human hand video to a dexterous hand or even a simple robot gripper (see Appendix D.6). In this section, however, we primarily focus on one type of transfer: human hand video to dexterous robot hand execution, as the majority of our robot experiments are done with a humanoid robot with a dexterous hand.

Human Hand Motion Estimation. We reconstruct an accurate and temporally aligned hand pose as first step for motion retargeting. To do so, we first predict hand pose in each video frame independently using image-based hand reconstruction model: *HaMeR* [62], then align and refine the predicted human hand with a dynamic scene reconstruction model, *MegaSAM* [50].

Per-frame Hand Pose Estimation. For each input frame I_t , *HaMeR* predicts MANO [67] hand vertices \mathbf{V}_t and a global wrist orientation $\mathbf{R}_t \in \text{SO}(3)$ in the camera coordinate frame. While *HaMeR* provides accurate hand shape and articulation, its per-frame translation estimates tend to drift over time due to the lack of temporal consistency enforcement.

4D Consistent Alignment. We then align the translations of the per-frame reconstructed human hand. Specifically, we leverage a 4D reconstruction model, *MegaSAM* [50], which outputs per-frame depth maps $D_t(u, v)$, camera intrinsics \mathbf{K} , and extrinsics $\{\mathbf{E}_t\}_{t=0}^{T-1}$. After getting per-frame depth and camera pose, we backproject pixels of the hand into 3D, where pixels of the hand (u_t, v_t) are obtained by projecting the MANO wrist joint regressed from \mathbf{V}_t .

We retain HaMeR’s orientation \mathbf{R}_t while using the backprojected wrist pointclouds to estimate \mathbf{T}_t . This enforces temporal smoothness, resolves monocular scale ambiguity, and significantly improves wrist localization robustness.

Temporal Completion and Smoothing. Frames with invalid depth/pixels are marked missing and linearly interpolated in position. Quaternions use SLERP with sign flips to maintain continuity. We then apply a causal Savitzky-Golay filter (window w , order d) to positions and quaternion components, followed by re-normalization, noted as $\hat{\mathbf{T}}_{\mathcal{R} \leftarrow \mathcal{W}, t}$.

Robot Finger Motion Retargeting. Given the human hand pose estimated by the previous module, we design retargeting modules that support both multi-finger dexterous hands and parallel-jaw grippers. We introduce multi-finger dexterous hands below and parallel-jaw grippers in Appendix D.6.

To retarget robot finger joints from human hands, we use *Dex-Retargeting* [66], which first extracts human hand keypoints using an RGB-based detector and then maps them to robot joint configurations by solving a DexPilot-style optimization objective. This produces robot finger joint angles $\mathbf{q}_t^R \in \mathbb{R}^{n_{\text{dof}}}$, enabling fine-grained imitation of articulated human manipulation.

We export per-frame wrist SE(3) $(\hat{\mathbf{p}}_{\mathcal{W}, t}^R, \hat{\mathbf{q}}_{\mathcal{W}, t}^R, \hat{\mathbf{T}}_{\mathcal{R} \leftarrow \mathcal{W}, t})$ and robot joints $\{\mathbf{q}_t^R\}$, together with metadata (joint names, DOF). Qualitative checks are performed by rendering the robot hand motions in simulation, see videos in the project website.

Real-Robot Execution. Given the human wrist trajectories $\{\mathbf{P}_t\}_{t=0}^{T-1}$ and the robot finger joint trajectories $\{\mathbf{q}_t\}_{t=0}^{T-1}$ estimated by the preceding modules (both expressed in the camera coordinates of the first video frame), our goal is to execute the motion on a physical robot. We first rotate the wrist poses into the robot control frame, then use the resulting wrist translations and orientations to solve the inverse kinematics (IK) for the arm (using cuRobo[74]), while the finger trajectories directly drive the robot hand joints.

Camera-to-Robot Alignment. Let \mathcal{C}_0 denote the coordinate frame of the first camera, \mathcal{M} the MANO hand frame, and \mathcal{R} the robot control frame. We align the recovered wrist poses from \mathcal{C}_0 to \mathcal{R} via an extrinsic calibration. In practice, this reduces to applying a fixed rotation $\mathbf{M} \in \text{SO}(3)$ that unifies the axes of \mathcal{C}_0 and \mathcal{R} :

$$\mathbf{p}_{\mathcal{W}, t}^R = \mathbf{M} \mathbf{p}_{\mathcal{W}, t}^{\mathcal{C}_0} + t, \quad \mathbf{R}_{\mathcal{W}, t}^R = \mathbf{M} \mathbf{R}_{\mathcal{W}, t}^{\mathcal{C}_0} \quad (5)$$

where $\mathbf{p}_{\mathcal{W}, t}^{\mathcal{C}_0}$ and $\mathbf{R}_{\mathcal{W}, t}^{\mathcal{C}_0}$ are the wrist translation and rotation at time t in \mathcal{C}_0 . We then assemble the wrist pose $\mathbf{T}_{\mathcal{R} \leftarrow \mathcal{W}, t} \in \text{SE}(3)$ and its quaternion parameterization $\mathbf{q}_{\mathcal{W}, t}^R$ for downstream control.

Robot Wrist and Finger Execution. With the wrist trajectory expressed in \mathcal{R} , we use cuRobo[74] to solve IK and obtain arm joint trajectories that follow $\{\mathbf{p}_{\mathcal{W}, t}^R, \mathbf{R}_{\mathcal{W}, t}^R\}_{t=0}^{T-1}$. In parallel, the finger joint sequence $\{\mathbf{q}_t\}_{t=0}^{T-1}$ is sent directly to the robot hand controller. Finally, the robot control API executes these synchronized arm and hand trajectories to complete the task.

4 EVALUATING TASK-LEVEL GENERALIZATION

Traditionally, robot foundation models are evaluated on tasks similar to those in their training sets - picking up slightly different objects at randomized locations after seeing a lot of pick-and-place trajectories, or folding t-shirts after learning from large amount of t-shirt folding data [12, 43, 83]. We refer to these as object-level and configuration-level generalizations. They are exciting steps towards a zero-shot model but the “verbs” in the task description are constrained to this small set, like “pick” or “fold”.

We are interested in a stronger type of generalization - zero-shot task-level generalization: evaluating whether a model can perform drastically different tasks it has never encountered.



(a) Pull out the tissue

(b) Open gate

Figure 5: **Baseline Comparison.** LVP accurately generates videos of hand interactions in a zero-shot setting, such as pulling out a tissue (left) and opening a gate (right). Baseline models (Wan, Cosmos-Predict 2, Hunyuan) often produce spatial or semantic inconsistencies, highlighted by red circles. The first frame and task instruction shown under each column serve as the generation conditions.

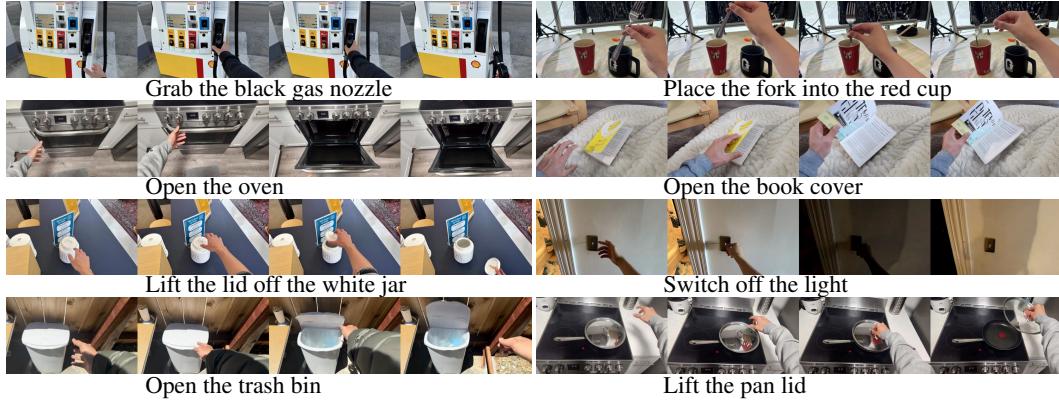


Figure 6: **Visualization of generated video plans.** Eight examples in our in-the-wild test set with generated videos by LVP.

4.1 THIRD-PARTY SELECTION OF NOVEL TASKS

We believe that true task-level generalization should allow any human to propose a task in any environment—without requiring prior knowledge of the capability of the model. To this end, we crowdsource test data from third-party participants by asking them to propose manipulation tasks from their everyday surroundings. Each participant was instructed to: (1) propose a short manipulation task that takes 3–5 seconds for a human; (2) take a photo of the scene showing both the hand and the target object; (3) write a brief text description of the intended task; and (4) stay diverse and challenging, be creative about tasks and scenes.

After this step, we gathered around 200 tasks with very out-of-distribution scene like “at a gasoline pump”, out-of-distribution yet hard tasks such as “flush the toilet” or “tear the tape”. However, we noticed that some volunteers still submitted low quality data such as blurry photos or boring tasks. To ensure quality, a separate group of third-party annotators filtered out samples that did not follow instructions or resembled basic tabletop pick-and-push tasks already covered in existing robot datasets. After filtering, 100 high-quality tasks remained, each consisting of one observation image and an instruction text. The instruction texts were further refined using Gemini to produce more detailed task descriptions.

4.2 EVALUATING VIDEO MOTION PLANNING

We first evaluate the stand-alone performance of our video planner. We feed observations and rephrased instruction texts to our model. Figure 6 shows qualitative examples of generated video plans on this in-the-wild test set. Figure 7 shows multi-stage video plans by extending generated videos repeatedly using our video-to-video generation.

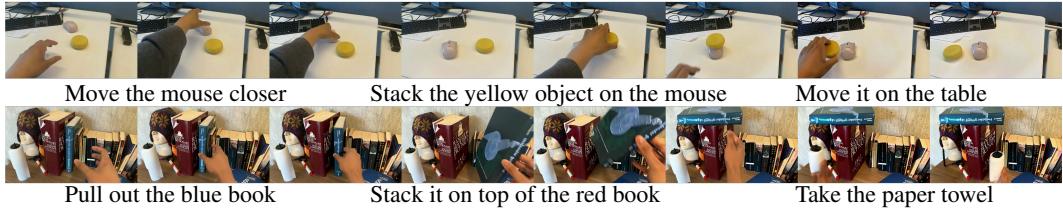


Figure 7: **Multi-Stage Video Plans.** Our LVP can generate long-horizon video plans by repeatedly extending videos conditioned on the last six latent frames. Each example illustrates a three-stage motion plan obtained through two iterative extensions.

Method	Level 1: Correct contact		Level 2: End state		Level 3: Task complete		Level 4: Perfect	
	Average (%)	Best@4	Average (%)	Best@4	Average (%)	Best@4	Average (%)	Best@4
Wan 2.1 I2V 14B	83.9	99.0	47.0	80.0	39.3	76.0	20.5	53.0
Cosmos-Predict 2 14B	45.3	81.0	11.9	35.0	7.5	24.0	2.5	9.0
Hunyuan I2V 14B	68.7	96.0	27.3	65.0	13.5	42.0	7.2	27.0
Ours	87.3	100.0	63.2	85.0	59.3	82.0	44.0	71.0

Table 2: **Video Plan Evaluation.** Evaluation on 100 in-the-wild manipulation prompts collected from third-party participants. We report the average success rate (Average) and Best@4 for each level. Our method achieves substantially higher success at Levels 3–4 than the baselines, indicating stronger generation of coherent, task-complete plans in in-the-wild settings.

We compare against three strong video generation baselines: Wan 2.1 I2V 14B [81], Cosmos-Predict 2 14B [2], and Hunyuan I2V 13B [45]. For each prompt, every method generates four videos.

We design a four-level evaluation metric that measures instruction following, motion planning feasibility, and physical realism.

- (1) *Correct contact*: The hand makes contact with the specified object at a correct location. Failures include touching the wrong object or making no contact.
- (2) *Correct end state*: The final frame achieves the instructed goal (motion quality ignored).
- (3) *Task complete*: Both correct contact and correct end state with plausible, continuous motion (minor physics artifacts allowed).
- (4) *Perfect task complete*: The task is completed with visually flawless physics and no noticeable artifacts. This highest level incorporates all prior criteria and additionally evaluates physical consistency and visual fidelity.

Levels 1–2 test comprehension and prompt following—whether the model correctly interprets and interacts with the right objects. Level 3 evaluates whether the model can generate complete video planning with feasible and coherent motions. Level 4 additionally measures physical realism and overall visual fidelity.

We ask third-party annotators to score all the generated videos and report both the average success rate and Best@4 (best result among four generations) for each level in Table 2, with quantitative comparisons illustrated in Figure 5.

For all models, performance decreases monotonically from Level 1 to Level 4, reflecting the increasing difficulty of each criterion. While pretrained Wan 2.1 achieves relatively high scores on Level 1 (correct contact), its performance drops sharply on Levels 2–4, indicating that it can initiate the correct interaction but fails to produce coherent, task-complete motion trajectories. In contrast, our model achieves significantly higher scores across all levels, with the largest gains at Levels 3 and 4, indicating better generalization in producing coherent, physically consistent motion planning under in-the-wild conditions. Notably, our model attains a 59.3% success rate at Level 3 (Task Complete) on the third-party test set, highlighting its ability to perform coherent and semantically grounded motion planning for unseen tasks in unseen environments. In addition, in Figure 7, we illustrate how our model is able to rollout long video plans.

Task Set and Tasks	Ours	π_0	OpenVLA
Task Group A: w/ Parallel Gripper			
Pick Objects	5/10	3/10	0/10
Pick A into B	3/10	1/10	0/10
Open Drawer	2/10	1/10	0/10
Press Button	4/10	0/10	0/10
Task Group B: w/ Dexterous Hands			
Pick Objects	4/10	N/A	N/A
Press Elevator Button	4/5	N/A	N/A
Sweep Tennis Ball into Bucket	5/5	N/A	N/A
Open Box (a)	2/10	N/A	N/A
Open Door (b)	6/10	N/A	N/A
Wipe Table (c)	8/10	N/A	N/A
Scoop Coffee Beans (b)	3/5	N/A	N/A
Tear off Clear Tape (b)	2/5	N/A	N/A
Task Group C: Out-of-distribution Set			
Pick Objects (OOD Object ¹)	4/10	0/10	0/10
Pick A into B (OOD Object ¹)	2/10	0/10	0/10
Pick Objects (OOD Scene ²)	6/10	1/10	0/10
Pick A into B (OOD Scene ²)	1/10	0/10	0/10

(a) Open the first-aid box

(b) Open the fridge door

(c) Wipe down the table and clean up the coffee stains

(d) Scoop coffee beans from the plate

(e) Tear off the clear tape

Figure 8: **Robot Execution Evaluation.** Left: Comparison of Task Success Across Methods on (1) Franka Arm with Parallel-Jaw Gripper and (2) G1 with Inspire Hands. ¹ denotes tests on OOD objects; ² denotes scenes that differ substantially from the training videos. Right: Visualization of the robot tasks and experiments.

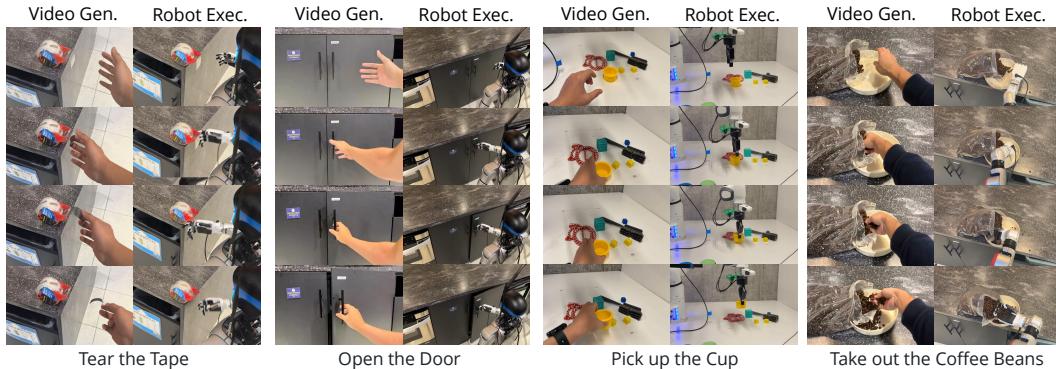


Figure 9: **Zero Shot Robot Manipulation with LVP.** The model generates videos for diverse tasks, enabling zero-shot execution on both a dexterous hand and a parallel-jaw gripper.

4.3 EVALUATING REAL-WORLD ROBOT MANIPULATION

The previous experiment demonstrates that our large video planner exhibits strong zero-shot generalization for unseen tasks and novel scenes. We now evaluate the complete pipeline, from video generation to action retargeting and execution, on real-world robotic platforms.

Tasks. We conduct experiments on two distinct robot morphologies: a Franka Emika Arm with a parallel-jaw gripper and a G1 Arm equipped with an Inspire dexterous hand. Each platform is tested on task sets that highlight different manipulation capabilities. For the dexterous hand, we further evaluate challenging novel tasks such as opening a door, opening a box, and scooping coffee beans, as shown in the right columns of Figure 8.

- *Franka Arm with Parallel-Jaw Gripper* As reported in Task Groups A and C of Table 8, this set focuses on manipulation tasks that can be achieved with simple two-finger grasps, such as object pick-and-place, block stacking, and bottle relocation. These tasks emphasize grasp detection and robust trajectory execution under limited actuation. Group A serves as the standard benchmark, while Group C primarily focuses on out-of-distribution task sets that contain unseen scenarios in the training video dataset. We include more details about gripper cases in Appendix D.6.
- *Humanoid with Dexterous Hand* As reported in Task Group B of Table 8, this set targets fine-grained dexterous manipulation requiring multiple degrees of freedom. Tasks include in-hand rotation, tool use (e.g., pen writing or screwdriver insertion), and precise placement of irregular objects. These tasks stress the ability of our method to transfer complex human hand articulations to the robot hand.

Baselines. We compare our method against several state-of-the-art vision-language-action baselines:

-
- π_0 [10]: We evaluate π_0 model by loading the released checkpoint and directly testing its generalization to our benchmark tasks, following the standard usage protocol.
 - *OpenVLA* [43]: We include OpenVLA with its released checkpoint as a baseline, evaluating its performance on our task sets without additional fine-tuning.

Note that π_0 and OpenVLA are not compatible with multi-finger dexterous hand settings and are therefore only tested on parallel-gripper tasks.

Results. Quantitative comparisons are presented in Table 8, and qualitative examples of successful executions are shown in Figure 9. Our approach exhibits strong zero-shot generalization on the most challenging settings—e.g., scooping coffee beans and tearing tape—underscoring the significance of the proposed method. Across both task suites, it consistently outperforms existing baselines, with especially outstanding performance on dexterous manipulation. In contrast, baselines show strong performance at tasks similar to training distributions, e.g. picking up objects, but struggle with task-level generalization. We speculate that such regression arises because imitation learning based robot foundation models have seen a lot of trajectories of pick-and-place, but have never seen enough diverse tasks to robustly generalize to new ones.

5 LIMITATIONS

Our approach has several limitations. On the video generation side, producing a single video plan takes several minutes on a single A100 GPU, making direct real-time deployment on robots intractable. Potential solutions include step-distillation methods [69, 86], which reduce the number of inference steps, or causal video models [16, 87, 41], which lower the latency of the generation process. In addition, on the robotics side, our robot action extraction has several limitations. Our current robotics action extraction pipeline uses open-source models to estimate 4D reconstructions and hand pose estimations. Both of these models can make mistakes, sometimes leading to task failures. Even if all models succeed, the retarget might not be sufficient for certain dexterous hands. In addition, retargeting actions to the parallel-jaw gripper can be challenging due to its much lower degree-of-freedom count compared to a human hand. Finally, our overall robot execution framework is run in an open-loop manner, which is not sufficient for accomplishing dexterous tasks.

6 CONCLUSION

We investigate a different approach to robot foundation models with video as the backbone. We present Large Video Planner (LVP), a 14-billion parameter video foundation model for embodiment planning. LVP generates videos as motion plans conditioned on one or a few scene frames and a text description of the task. We demonstrate that these generated motion plans can be successfully retargeted to dexterous robotic hands using open-source reconstruction and retargeting tools. Evaluations on third-party proposed tasks show evidence of task-level generalization, a capability limited in existing VLA models. We open-source our model, data, and training code to support the research community and hope this work will inspire further exploration of video foundation model for robotics.

BIBLIOGRAPHY

- [1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. [1](#)
- [2] Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*, 2025. [3](#), [11](#)
- [3] Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Daniel Ho, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Eric Jang, Rosario Jauregui Ruano, Kyle Jeffrey, Sally Jesmonth, Nikhil J Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Kuang-Huei Lee, Sergey Levine, Yao Lu, Linda Luu, Carolina Parada, Peter Pastor, Jornell Quiambao, Kanishka Rao, Jarek Rettinghouse, Diego Reyes, Pierre Sermanet, Nicolas Sievers, Clayton Tan, Alexander Toshev, Vincent Vanhoucke, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Mengyuan Yan, and Andy Zeng. Do as i can, not as i say: Grounding language in robotic affordances, 2022. URL <https://arxiv.org/abs/2204.01691>. [3](#)
- [4] Anurag Ajay, Seungwook Han, Yilun Du, Shuang Li, Abhi Gupta, Tommi Jaakkola, Josh Tenenbaum, Leslie Kaelbling, Akash Srivastava, and Pulkit Agrawal. Compositional foundation models for hierarchical planning. *Advances in Neural Information Processing Systems*, 36: 22304–22325, 2023. [3](#)
- [5] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023. [1](#)
- [6] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. [1](#)
- [7] Jose Barreiros, Andrew Beaulieu, Aditya Bhat, Rick Cory, Eric Cousineau, Hongkai Dai, Ching-Hsin Fang, Kuniatsu Hashimoto, Muhammad Zubair Irshad, Masha Itkina, et al. A careful examination of large behavior models for multitask dexterous manipulation. *arXiv preprint arXiv:2507.05331*, 2025. [3](#)
- [8] Homanga Bharadhwaj, Debidatta Dwibedi, Abhinav Gupta, Shubham Tulsiani, Carl Doersch, Ted Xiao, Dhruv Shah, Fei Xia, Dorsa Sadigh, and Sean Kirmani. Gen2act: Human video generation in novel scenarios enables generalizable robot manipulation. *arXiv preprint arXiv:2409.16283*, 2024. [3](#)
- [9] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0 : A vision-language-action flow model for general robot control, 2024. URL <https://arxiv.org/abs/2410.24164>. [3](#)
- [10] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. pi_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024. [1](#), [13](#)
- [11] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. [3](#)
- [12] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022. [1](#), [9](#)

-
- [13] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alexander Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Lisa Lee, Tsang-Wei Edward Lee, Sergey Levine, Yao Lu, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael Ryoo, Grecia Salazar, Pannag Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricu, Huong Tran, Vincent Vanhoucke, Quan Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Jialin Wu, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. Rt-2: Vision-language-action models transfer web knowledge to robotic control, 2023. URL <https://arxiv.org/abs/2307.15818>. 3
- [14] Tim Brooks, Bill Peebles, Connor Homes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. URL <https://openai.com/research/video-generation-models-as-world-simulators>. 3, 4
- [15] Qingwen Bu, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Shenyuan Gao, Xindong He, Xuan Hu, Xu Huang, et al. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. *arXiv preprint arXiv:2503.06669*, 2025. 1, 7
- [16] Boyuan Chen, Diego Marti Monso, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 2, 3, 4, 5, 13
- [17] Hanzhi Chen, Boyang Sun, Anran Zhang, Marc Pollefeys, and Stefan Leutenegger. Vidbot: Learning generalizable 3d actions from in-the-wild 2d human videos for zero-shot robotic manipulation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 27661–27672, 2025. 3
- [18] Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang-wei Chao, Byung Eun Jeon, Yuwei Fang, Hsin-Ying Lee, Jian Ren, Ming-Hsuan Yang, et al. Panda-70m: Captioning 70m videos with multiple cross-modality teachers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13320–13331, 2024. 1, 6, 22
- [19] Hyung Won Chung, Noah Constant, Xavier Garcia, Adam Roberts, Yi Tay, Sharan Narang, and Orhan Firat. Unimax: Fairer and more effective language sampling for large-scale multilingual pretraining. *arXiv preprint arXiv:2304.09151*, 2023. 5
- [20] Open X-Embodiment Collaboration, Abby O'Neill, Abdul Rehman, Abhinav Gupta, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, Albert Tung, Alex Bewley, Alex Herzog, Alex Irpan, Alexander Khazatsky, Anant Rai, Anchit Gupta, Andrew Wang, Andrey Kolobov, Anikait Singh, Animesh Garg, Aniruddha Kembhavi, Annie Xie, Anthony Brohan, Antonin Raffin, Archit Sharma, Arefeh Yavary, Arhan Jain, Ashwin Balakrishna, Ayzaan Wahid, Ben Burgess-Limerick, Beomjoon Kim, Bernhard Schölkopf, Blake Wulfe, Brian Ichter, Cewu Lu, Charles Xu, Charlotte Le, Chelsea Finn, Chen Wang, Chenfeng Xu, Cheng Chi, Chenguang Huang, Christine Chan, Christopher Agia, Chuer Pan, Chuyuan Fu, Coline Devin, Danfei Xu, Daniel Morton, Danny Driess, Daphne Chen, Deepak Pathak, Dhruv Shah, Dieter Büchler, Dinesh Jayaraman, Dmitry Kalashnikov, Dorsa Sadigh, Edward Johns, Ethan Foster, Fangchen Liu, Federico Ceola, Fei Xia, Feiyu Zhao, Felipe Vieira Frujeri, Freek Stulp, Gaoyue Zhou, Gaurav S. Sukhatme, Gautam Salhotra, Ge Yan, Gilbert Feng, Giulio Schiavi, Glen Berseth, Gregory Kahn, Guangwen Yang, Guanzhi Wang, Hao Su, Hao-Shu Fang, Haochen Shi, Henghui Bao, Heni Ben Amor, Henrik Christensen, Hiroki Furuta, Homanga Bharadhwaj, Homer Walke, Hongjie Fang, Huy Ha, Igor Mordatch, Ilija Radosavovic, Isabel Leal, Jacky Liang, Jad Abou-Chakra, Jaehyung Kim, Jaimyn Drake, Jan Peters, Jan Schneider, Jasmine Hsu, Jay Vakil, Jeannette Bohg, Jeffrey Bingham, Jeffrey Wu, Jensen Gao, Jiaheng Hu, Jiajun Wu, Jialin Wu, Jiankai Sun, Jianlan Luo, Jiayuan Gu, Jie Tan, Jihoon Oh, Jimmy Wu, Jingpei Lu, Jingyun Yang, Jitendra Malik, João Silvério, Joey Hejna, Jonathan Booher, Jonathan Tompson, Jonathan Yang, Jordi Salvador, Joseph J. Lim, Junhyek Han, Kaiyuan Wang, Kanishka Rao, Karl Pertsch, Karol Hausman, Keegan Go, Keerthana Gopalakrishnan, Ken Goldberg, Kendra Byrne, Kenneth Oslund, Kento

Kawaharazuka, Kevin Black, Kevin Lin, Kevin Zhang, Kiana Ehsani, Kiran Lekkala, Kirsty Ellis, Krishan Rana, Krishnan Srinivasan, Kuan Fang, Kunal Pratap Singh, Kuo-Hao Zeng, Kyle Hatch, Kyle Hsu, Laurent Itti, Lawrence Yunliang Chen, Lerrel Pinto, Li Fei-Fei, Liam Tan, Linxi "Jim" Fan, Lionel Ott, Lisa Lee, Luca Weihs, Magnum Chen, Marion Lepert, Marius Memmel, Masayoshi Tomizuka, Masha Itkina, Mateo Guaman Castro, Max Spero, Maximilian Du, Michael Ahn, Michael C. Yip, Mingtong Zhang, Mingyu Ding, Minho Heo, Mohan Kumar Srirama, Mohit Sharma, Moo Jin Kim, Muhammad Zubair Irshad, Naoaki Kanazawa, Nicklas Hansen, Nicolas Heess, Nikhil J Joshi, Niko Suenderhauf, Ning Liu, Norman Di Palo, Nur Muhammad Mahi Shafullah, Oier Mees, Oliver Kroemer, Osbert Bastani, Pannag R Sanketi, Patrick "Tree" Miller, Patrick Yin, Paul Wohlhart, Peng Xu, Peter David Fagan, Peter Mitrano, Pierre Sermanet, Pieter Abbeel, Priya Sundaresan, Qiyu Chen, Quan Vuong, Rafael Rafailov, Ran Tian, Ria Doshi, Roberto Mart'in-Mart'in, Rohan Baijal, Rosario Scalise, Rose Hendrix, Roy Lin, Runjia Qian, Ruohan Zhang, Russell Mendonca, Rutav Shah, Ryan Hoque, Ryan Julian, Samuel Bustamante, Sean Kirmani, Sergey Levine, Shan Lin, Sherry Moore, Shikhar Bahl, Shivin Dass, Shubham Sonawani, Shubham Tulsiani, Shuran Song, Sichun Xu, Siddhant Haldar, Siddharth Karamchetti, Simeon Adegbola, Simon Guist, Soroush Nasiriany, Stefan Schaal, Stefan Welker, Stephen Tian, Subramanian Ramamoorthy, Sudeep Dasari, Suneel Belkhale, Sungjae Park, Suraj Nair, Suvir Mirchandani, Takayuki Osa, Tanmay Gupta, Tatsuya Harada, Tatsuya Matsushima, Ted Xiao, Thomas Kollar, Tianhe Yu, Tianli Ding, Todor Davchev, Tony Z. Zhao, Travis Armstrong, Trevor Darrell, Trinity Chung, Vidhi Jain, Vikash Kumar, Vincent Vanhoucke, Vitor Guizilini, Wei Zhan, Wenxuan Zhou, Wolfram Burgard, Xi Chen, Xiangyu Chen, Xiaolong Wang, Xinghao Zhu, Xinyang Geng, Xiyuan Liu, Xu Liangwei, Xuanlin Li, Yansong Pang, Yao Lu, Yecheng Jason Ma, Yejin Kim, Yevgen Chebotar, Yifan Zhou, Yifeng Zhu, Yilin Wu, Ying Xu, Yixuan Wang, Yonatan Bisk, Yongqiang Dou, Yoonyoung Cho, Youngwoon Lee, Yuchen Cui, Yue Cao, Yueh-Hua Wu, Yujin Tang, Yuke Zhu, Yunchu Zhang, Yunfan Jiang, Yunshuang Li, Yunzhu Li, Yusuke Iwasawa, Yutaka Matsuo, Zehan Ma, Zhuo Xu, Zichen Jeff Cui, Zichen Zhang, Zipeng Fu, and Zipeng Lin. Open X-Embodiment: Robotic learning datasets and RT-X models. <https://arxiv.org/abs/2310.08864>, 2023. 1, 3

- [21] Gheorghe Comanici, Eric Bieber, Mike Schaeckermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025. 1
- [22] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Jian Ma, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision: Collection, pipeline and challenges for epic-kitchens-100. *International Journal of Computer Vision (IJCV)*, 130:33–55, 2022. URL <https://doi.org/10.1007/s11263-021-01531-2>. 6
- [23] Yufei Ding, Haoran Geng, Chaoyi Xu, Xiaomeng Fang, Jiazhao Zhang, Songlin Wei, Qiyu Dai, Zhizheng Zhang, and He Wang. Open6DOR: Benchmarking open-instruction 6-dof object rearrangement and a vlm-based approach. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 7359–7366, 2024. doi: 10.1109/IROS58592.2024.10802733. 3
- [24] Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. PALM-E: An embodied multimodal language model, 2023. URL <https://arxiv.org/abs/2303.03378>. 3
- [25] Yilun Du, Mengjiao Yang, Pete Florence, Fei Xia, Ayzaan Wahid, Brian Ichter, Pierre Sermanet, Tianhe Yu, Pieter Abbeel, Joshua B Tenenbaum, et al. Video language planning. *arXiv preprint arXiv:2310.10625*, 2023. 3
- [26] Yilun Du, Sherry Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Josh Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. *Advances in neural information processing systems*, 36:9156–9172, 2023. 2, 3

-
- [27] Frederik Ebert, Yanlai Yang, Karl Schmeckpeper, Bernadette Bucher, Georgios Georgakis, Kostas Daniilidis, Chelsea Finn, and Sergey Levine. Bridge data: Boosting generalization of robotic skills with cross-domain datasets. *arXiv preprint arXiv:2109.13396*, 2021. 3
- [28] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024. 4
- [29] Hao-Shu Fang, Chenxi Wang, Minghao Gou, and Cewu Lu. Graspnet: A large-scale clustered and densely annotated dataset for object grasping, 2020. URL <https://arxiv.org/abs/1912.13470>. 25
- [30] Caelan Reed Garrett, Rohan Chitnis, Rachel Holladay, Beomjoon Kim, Tom Silver, Leslie Pack Kaelbling, and Tomás Lozano-Pérez. Integrated task and motion planning. *Annual review of control, robotics, and autonomous systems*, 4(1):265–293, 2021. 1
- [31] Haoran Geng, Songlin Wei, Congyue Deng, Bokui Shen, He Wang, and Leonidas Guibas. Sage: Bridging semantic and actionable parts for generalizable articulated-object manipulation under language instructions, 2023. 3
- [32] Haoran Geng, Feishi Wang, Songlin Wei, Yuyang Li, Bangjun Wang, Boshi An, Charlie Tianyue Cheng, Haozhe Lou, Peihao Li, Yen-Jen Wang, Yutong Liang, Dylan Goetting, Chaoyi Xu, Haozhe Chen, Yuxi Qian, Yiran Geng, Jiageng Mao, Weikang Wan, Mingtong Zhang, Jiangran Lyu, Siheng Zhao, Jiazhao Zhang, Jialiang Zhang, Chengyang Zhao, Haoran Lu, Yufei Ding, Ran Gong, Yuran Wang, Yuxuan Kuang, Ruihai Wu, Baoxiong Jia, Carlo Sferrazza, Hao Dong, Siyuan Huang, Yue Wang, Jitendra Malik, and Pieter Abbeel. Roboverse: Towards a unified platform, dataset and benchmark for scalable and generalizable robot learning, 2025. URL <https://arxiv.org/abs/2504.18904>. 3
- [33] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The "something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision*, pp. 5842–5850, 2017. 6
- [34] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18995–19012, 2022. 6
- [35] Junlin Han, Shengbang Tong, David Fan, Yufan Ren, Koustuv Sinha, Philip Torr, and Filippos Kokkinos. Learning to see before seeing: Demystifying llm visual priors from language pre-training. *arXiv preprint arXiv:2509.26625*, 2025. 1
- [36] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 3, 5
- [37] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 3, 4
- [38] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 3
- [39] Yining Hong, Beide Liu, Maxine Wu, Yuanhao Zhai, Kai-Wei Chang, Linjie Li, Kevin Lin, Chung-Ching Lin, Jianfeng Wang, Zhengyuan Yang, et al. Slowfast-vgen: Slow-fast learning for action-driven long video generation. *arXiv preprint arXiv:2410.23277*, 2024. 3
- [40] Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. VoxPoser: Composable 3D value maps for robotic manipulation with language models, 2023. URL <https://arxiv.org/abs/2307.05973>. 3

-
- [41] Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *arXiv preprint arXiv:2506.08009*, 2025. 3, 13
- [42] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024. 3, 7
- [43] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An open-source vision-language-action model, 2024. URL <https://arxiv.org/abs/2406.09246>. 1, 3, 9, 13
- [44] Po-Chen Ko, Jiayuan Mao, Yilun Du, Shao-Hua Sun, and Joshua B Tenenbaum. Learning to act from actionless videos through dense correspondences. *arXiv preprint arXiv:2310.08576*, 2023. 3
- [45] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. HunyuanVideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024. 6, 11
- [46] Peihao Li, Haoran Geng, Jameson Crate, Yanbing Han, Junyi Zhang, Feishi Wang, Charlie Tianyue Cheng, Runpei Dong, Yen-Jen Wang, Haozhe Lou, et al. ROSE: Reconstructing objects, scenes, and trajectories from casual videos for robotic manipulation. In *Human to Robot: Workshop on Sensorizing, Modeling, and Learning from Humans*. 3
- [47] Shuang Li, Yihuai Gao, Dorsa Sadigh, and Shuran Song. Unified video action model. *arXiv preprint arXiv:2503.00200*, 2025. 3
- [48] Xiaoqi Li, Mingxu Zhang, Yiran Geng, Haoran Geng, Yuxing Long, Yan Shen, Renrui Zhang, Jiaming Liu, and Hao Dong. ManipLLM: Embodied multimodal large language model for object-centric robotic manipulation, 2023. 3
- [49] Xinghang Li, Minghuan Liu, Hanbo Zhang, Cunjun Yu, Jie Xu, Hongtao Wu, Chilam Cheang, Ya Jing, Weinan Zhang, Huaping Liu, Hang Li, and Tao Kong. Vision-language foundation models as effective robot imitators. *arXiv preprint arXiv:2311.01378*, 2023. 3
- [50] Zhengqi Li, Richard Tucker, Forrester Cole, Qianqian Wang, Linyi Jin, Vickie Ye, Angjoo Kanazawa, Aleksander Holynski, and Noah Snavely. MegaSaM: Accurate, fast, and robust structure and motion from casual dynamic videos, 2024. URL <https://arxiv.org/abs/2412.04463>. 3, 8, 9, 23
- [51] Junbang Liang, Ruoshi Liu, Ege Ozguroglu, Sruthi Sudhakar, Achal Dave, Pavel Tokmakov, Shuran Song, and Carl Vondrick. Dreamitate: Real-world visuomotor policy learning via video generation. *arXiv preprint arXiv:2406.16862*, 2024. 3
- [52] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022. 3, 4
- [53] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024. 1
- [54] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 1
- [55] YuXuan Liu, Abhishek Gupta, Pieter Abbeel, and Sergey Levine. Imitation from observation: Learning to imitate behaviors from raw video via context translation. In *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 1118–1125. IEEE, 2018. 3
- [56] Calvin Luo, Zilai Zeng, Yilun Du, and Chen Sun. Solving new tasks by adapting internet video knowledge. *arXiv preprint arXiv:2504.15369*, 2025. 3

-
- [57] Yunhao Luo and Yilun Du. Grounding video models to actions through goal conditioned exploration. *arXiv preprint arXiv:2411.07223*, 2024. 3
- [58] Corey Lynch, Ayzaan Wahid, Jonathan Tompson, Tianli Ding, James Betker, Robert Baruch, Travis Armstrong, and Pete Florence. Interactive language: Talking to robots in real time. *IEEE Robotics and Automation Letters*, 2023. 7
- [59] Robert McCarthy, Daniel CH Tan, Dominik Schmidt, Fernando Acero, Nathan Herr, Yilun Du, Thomas G Thuruthel, and Zhibin Li. Towards generalist robot learning from internet video: A survey. *Journal of Artificial Intelligence Research*, 83, 2025. 3
- [60] Jack Monas and Eric Jang. 1x world model. Blog post, 1X Discover, September 2024. URL <https://www.1x.tech/discover/1x-world-model>. Accessed: 2025-09-22. 3
- [61] NVIDIA, Nikita Cherniadev, Johan Bjorck and Fernando Castañeda, Xingye Da, Runyu Ding, Linxi "Jim" Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, Joel Jang, Zhenyu Jiang, Jan Kautz, Kaushil Kundalia, Lawrence Lao, Zhiqi Li, Zongyu Lin, Kevin Lin, Guilin Liu, Edith Llontop, Loic Magne, Ajay Mandlekar, Avnish Narayan, Soroush Nasiriany, Scott Reed, You Liang Tan, Guanzhi Wang, Zu Wang, Jing Wang, Qi Wang, Jiannan Xiang, Yuqi Xie, Yinzheng Xu, Zhenjia Xu, Seonghyeon Ye, Zhiding Yu, Ao Zhang, Hao Zhang, Yizhou Zhao, Ruijie Zheng, and Yuke Zhu. GR00T N1: An open foundation model for generalist humanoid robots. In *ArXiv Preprint*, March 2025. 3
- [62] Georgios Pavlakos, Dandan Shan, Ilija Radosavovic, Angjoo Kanazawa, David Fouhey, and Jitendra Malik. Reconstructing hands in 3D with transformers. In *CVPR*, 2024. 3, 8, 23
- [63] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023. 4
- [64] Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Colin A Raffel, Leandro Von Werra, Thomas Wolf, et al. The fineweb datasets: Decanting the web for the finest text data at scale. *Advances in Neural Information Processing Systems*, 37:30811–30849, 2024. 1
- [65] Han Qi, Haocheng Yin, Aris Zhu, Yilun Du, and Heng Yang. Strengthening generative robot policies through predictive world modeling. *arXiv preprint arXiv:2502.00622*, 2025. 3
- [66] Yuzhe Qin, Wei Yang, Binghao Huang, Karl Van Wyk, Hao Su, Xiaolong Wang, Yu-Wei Chao, and Dieter Fox. Anyteleop: A general vision-based dexterous robot arm-hand teleoperation system. In *Robotics: Science and Systems*, 2023. 3, 9
- [67] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *arXiv preprint arXiv:2201.02610*, 2022. 8
- [68] David Ruhe, Jonathan Heek, Tim Salimans, and Emiel Hoogeboom. Rolling diffusion models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 42818–42835. PMLR, 21–27 Jul 2024. 3
- [69] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022. 13
- [70] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 35:25278–25294, 2022. 1
- [71] Lin Shao, Toki Migimatsu, Qiang Zhang, Karen Yang, and Jeannette Bohg. Concept2robot: Learning manipulation concepts from instructions and human demonstrations. *The International Journal of Robotics Research*, 40(12-14):1419–1434, 2021. 3

-
- [72] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pp. 2256–2265. pmlr, 2015. 3, 4
- [73] Kiwhan Song, Boyuan Chen, Max Simchowitz, Yilun Du, Russ Tedrake, and Vincent Sitzmann. History-guided video diffusion. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025. 2, 3, 4, 5
- [74] Balakumar Sundaralingam, Siva Kumar Sastry Hari, Adam Fishman, Caelan Garrett, Karl Van Wyk, Valts Blukis, Alexander Millane, Helen Oleynikova, Ankur Handa, Fabio Ramos, Nathan Ratliff, and Dieter Fox. curobo: Parallelized collision-free minimum-jerk robot motion generation, 2023. 9
- [75] Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024. 1
- [76] Gemini Robotics Team, Saminda Abeyruwan, Joshua Ainslie, Jean-Baptiste Alayrac, Montserrat Gonzalez Arenas, Travis Armstrong, Ashwin Balakrishna, Robert Baruch, Maria Bauza, Michiel Blokzijl, et al. Gemini robotics: Bringing ai into the physical world. *arXiv preprint arXiv:2503.20020*, 2025. 3
- [77] Kimi Team, Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, et al. Kimi k2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*, 2025. 1
- [78] Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, You Liang Tan, Lawrence Yunliang Chen, Pannag Sanketi, Quan Vuong, Ted Xiao, Dorsa Sadigh, Chelsea Finn, and Sergey Levine. Octo: An open-source generalist robot policy, 2024. URL <https://arxiv.org/abs/2405.12213>. 3
- [79] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 1
- [80] Homer Rich Walke, Kevin Black, Tony Z Zhao, Quan Vuong, Chongyi Zheng, Philippe Hansen-Estruch, Andre Wang He, Vivek Myers, Moo Jin Kim, Max Du, et al. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning*, pp. 1723–1736. PMLR, 2023. 3, 7
- [81] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. 3, 4, 5, 6, 11
- [82] Boyang Wang, Nikhil Sridhar, Chao Feng, Mark Van der Merwe, Adam Fishman, Nima Fazeli, and Jeong Joon Park. This&that: Language-gesture controlled video generation for robot planning. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 12842–12849. IEEE, 2025. 3
- [83] J. Wang, M. Leonard, K. Daniilidis, D. Jayaraman, and E. S. Hu. Evaluating pi0 in the wild: Strengths, problems, and the future of generalist robot policies, 2025. URL <https://penn-pal-lab.github.io/pi0-Experiment-in-the-Wild>. 1, 9
- [84] Haoyu Xiong, Quanzhou Li, Yun-Chun Chen, Homanga Bharadhwaj, Samarth Sinha, and Animesh Garg. Learning by watching: Physical imitation of manipulation skills from human videos. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 7827–7834, 2021. doi: 10.1109/IROS51168.2021.9636080. 3

-
- [85] Mengjiao Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. *arXiv preprint arXiv:2310.06114*, 1(2):6, 2023. 3
- [86] Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6613–6623, 2024. 13
- [87] Tianwei Yin, Qiang Zhang, Richard Zhang, William T Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 22963–22974, 2025. 3, 13
- [88] Lvmin Zhang, Shengqu Cai, Muyang Li, Gordon Wetzstein, and Maneesh Agrawala. Frame context packing and drift prevention in next-frame-prediction video diffusion models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*. 3
- [89] Haoyu Zhen, Qiao Sun, Hongxin Zhang, Junyan Li, Siyuan Zhou, Yilun Du, and Chuang Gan. Tesseract: learning 4d embodied world models. *arXiv preprint arXiv:2504.20995*, 2025. 3
- [90] Fangqi Zhu, Hongtao Wu, Song Guo, Yuxiao Liu, Chilam Cheang, and Tao Kong. Irasim: Learning interactive real-robot action simulators. *arXiv preprint arXiv:2406.14540*, 2024. 3
- [91] Minjie Zhu, Yichen Zhu, Jinming Li, Junjie Wen, Zhiyuan Xu, Ning Liu, Ran Cheng, Chaomin Shen, Yixin Peng, Feifei Feng, et al. Scaling diffusion policy in transformer to 1 billion parameters for robotic manipulation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 10838–10845. IEEE, 2025. 3

A DETAILS OF LVP-1M DATASET

A.1 EXTRACTING HAND INTERACTION VIDEOS FROM PANDAS

As summarized in Table 1, we extract a subset of 196K video clips focusing on human hand interactions from the Pandas-70M dataset [18]. The process is as follows.

We begin with the metadata from Pandas-70M, which provides short text captions for each clip. Using a whitelist of 108 keywords and a blacklist of 84 keywords, we perform an initial keyword-based filtering. After this step, we download approximately 692K videos of variable length.

Each video is segmented into multiple non-overlapping 4-second clips, and we run human pose detection at 1 FPS on frames resized to 768×1024 . We retain only clips containing 1–3 valid human detections, where a detection is considered valid if the bounding box height is ≤ 60 pixels and width is ≥ 120 pixels.

Next, we use Gemini-2.0 Flash to caption and evaluate each video based on four questions:

- (1) Does the video contain rich human hand motions?
- (2) Does the video show a human performing any meaningful actions (e.g., manipulating objects, using tools, cooking, pushing or pulling objects, waving, clapping, washing hands, etc.)?
- (3) Is the video playing at a normal speed?
- (4) Does the video contain scene changes (i.e., camera shot changes or viewpoint changes)?

We retain only videos with answers True, True, True, and False to these four questions. Finally, we retain 196K clips.

A.2 DATA PREPROCESSING PIPELINE

Here, we illustrate the data curation pipeline in Figure 10. We also illustrate the final composition of the LVP-1M using a pie-chart in Figure 11. We also include some details about each stage below.

Filtering our rapid camera motions. Lot’s of egocentric videos from Ego4D, Epic-Kitchens contains large camera motions, this hinders the model from learning forgound actions and we hope to remove majority of them. To do so, we compute per-pixel optical flow using OpenCV¹ at 4 frames per second on resized frames of 256×256 resolution, and remove the top 30% of videos with the highest spatio-temporal average flow magnitude.

B DETAILS OF VIDEO GENERATIVE MODEL TRAINING

Each training sample consists of a 49-frame video clip at a resolution of 832×480 . The clip is encoded into VAE latent at shape of $104 \times 60 \times 13$. In the continue pretraining stage, we train the model with batch size of 128 for 60,000 iterations, we use a constant learning rate of 1×10^{-5} after 1000 steps of warmup. During the finetuning stage, we use the same batch size and trains for 10,000 iterations with a reduced learning rate at 2.5×10^{-6} .

Data reweighting. As shown in Table 1, the eight dataset sources vary greatly in size. To balance their contributions during training, we employ weighted sampling with respective weights of 0.375, 0.75, 1.5, 0.5, 0.5, 1.0, 2.0, and 0.05 for AgiBot-World, DROID, Ego4D, Pandas (filtered), SomethingSomething, Bridge, Epic-Kitchens, and Language Table. The resulting sampling composition is visualized in Figure 11.

C METHOD: COMPARISON BETWEEN HAMER AND OUR 4D ALIGNMENT MODULE

In HaMeR, the depth estimation module lacks accuracy in metric scale estimation. Also, the hand hand pose is estimated with no temporal consistency. Hence, introducing MegaSaM would improve accuracy in depth estimation and smoothness in temporal hand trajectory estimation. To evaluate the effectiveness of our 4D Alignment module, we conduct an ablation study highlighting its ability to enforce spatio-temporal consistency, see Fig. 12.

¹<https://opencv.org/>

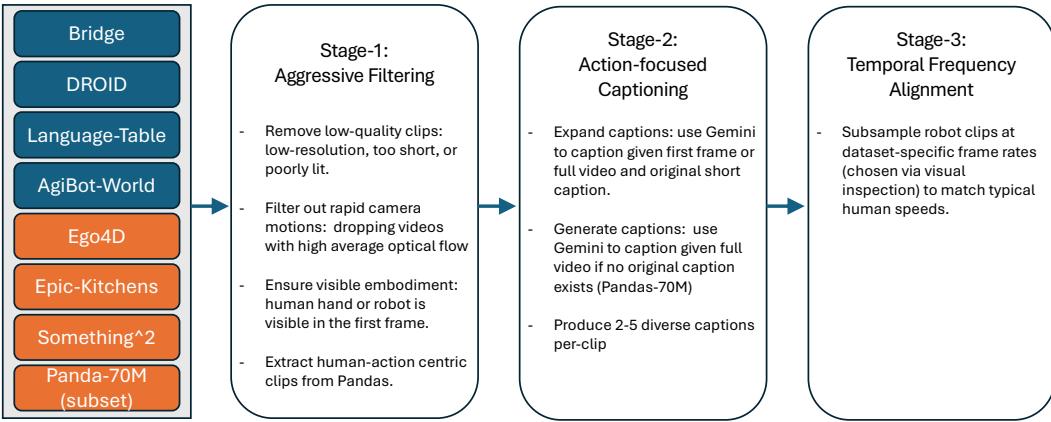


Figure 10: **LVP-1M curation pipelines.** Videos are collected from eight public sources, including four teleoperated robotics datasets and four human-centric activity datasets. We apply three processing stages: (1) aggressive filtering for quality and embodiment, (2) action-focused captioning using Gemini, and (3) temporal frequency alignment to match human motion speeds. The final dataset contains 1.4 million clips with diverse, action-centric text captions.

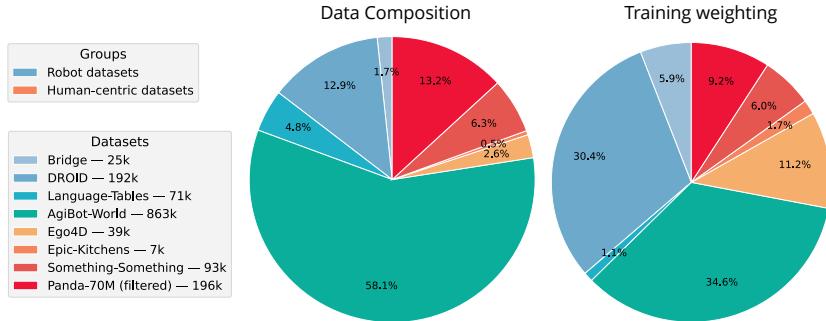


Figure 11: **Composition of LVP-1M .** Distribution of filtered LVP-1M clips across eight sources. Greenish tones indicate teleoperated robot datasets, while reddish tones represent human-centric activity datasets. The rightmost column shows the final sampling ratios after reweighting during first-stage training.

D METHOD: DETAILS ABOUT OUR VIDEO PROCESSING MODULES

D.1 DATA ACQUISITION AND INPUT

Given a monocular RGB sequence $\{I_t\}_{t=1}^T$ without known camera intrinsics or extrinsics, our objective is to recover a metrically consistent, temporally smooth hand motion representation that includes (i) per-frame joints angles and (ii) a wrist pose trajectory in a fixed world frame.

D.2 HAMER: HAND MESH AND CAMERA ESTIMATION

For each frame, HaMeR [62] predicts MANO parameters, hand mesh vertices, and weak-perspective camera parameters. These outputs provide articulated 3D joint positions and their 2D projections, together with a global wrist orientation and translation, forming the per-frame hand motion estimates. However, due to the inherent scale ambiguity of monocular weak-perspective modeling and the lack of temporal constraints, HaMeR’s predicted translations are often noisy and drift over time. As a result, the recovered wrist trajectories are not metrically accurate and cannot be directly used for real-world robot execution. To resolve these issues, we incorporate depth and multi-view cues from MegaSaM, which performs global optimization across frames to refine both the metric scale and temporal consistency of the estimated hand motion.

D.3 MEGASAM: DEPTH-ASSISTED STRUCTURE RECOVERY

To improve temporal consistency and resolve monocular scale ambiguity, we integrate MegaSaM [50]. Depth maps are estimated per frame and refined with bundle adjustment across the sequence, yielding

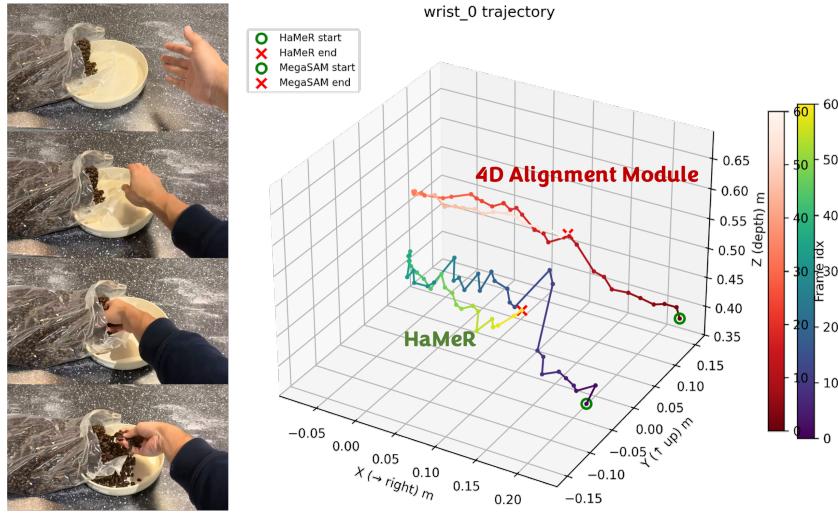


Figure 12: **Comparison between HaMeR and Our 4D Alignment Module.** The red curve shows our 4D Alignment Module; the blue one is from HaMeR.

per-frame extrinsics and metrically aligned depth. The first frame is chosen as the world reference system.

D.4 ALIGNMENT OF HAMER AND MEGASAM

We first use MegaSaM to estimate the intrinsic parameters \mathbf{K} , which are then provided to HaMeR for more reliable per-frame wrist localization. Specifically, HaMeR outputs the wrist pixel coordinate (u_t, v_t) in the image plane. We then sample the corresponding depth $z_t = D_t(u_t, v_t)$ from the MegaSaM depth map and back-project it using \mathbf{K} :

$$\mathbf{p}_{t,w}^{\text{cam}} = z_t \mathbf{K}^{-1} \begin{bmatrix} u_t \\ v_t \\ 1 \end{bmatrix}. \quad (6)$$

The wrist position in the first frame $\mathbf{p}_{t,w}^{\text{w}}$ is recovered by applying the optimized extrinsics $\mathbf{T}_t = [\mathbf{R}_t | \mathbf{t}_t]$ from MegaSaM:

$$\mathbf{p}_{t,w}^{\text{w}} = \mathbf{R}_t \mathbf{p}_{t,w}^{\text{cam}} + \mathbf{t}_t. \quad (7)$$

For Camera-to-Robot Alignment, we note the first camera frame \mathcal{C}_0 , the MANO frame \mathcal{M} , and the robot frame \mathcal{R} . We apply the rotation $\mathbf{M} \in \text{SO}(3)$ to unify axes between \mathcal{C}_0 and \mathcal{R} . Translations use the same linear map:

$$\mathbf{p}_{\mathcal{W},t}^{\mathcal{R}} = \mathbf{M} \mathbf{p}_{\mathcal{W},t}^{\mathcal{C}_0}, \quad \mathbf{R}_{\mathcal{W},t}^{\mathcal{R}} = \mathbf{M} \mathbf{R}_{\mathcal{W},t}^{\mathcal{C}_0} \mathbf{M}^\top. \quad (8)$$

We then form the wrist pose $\mathbf{T}_{\mathcal{R} \leftarrow \mathcal{W},t} = \begin{bmatrix} \mathbf{R}_{\mathcal{W},t}^{\mathcal{R}} & \mathbf{p}_{\mathcal{W},t}^{\mathcal{R}} \\ \mathbf{0}^\top & 1 \end{bmatrix}$ and its quaternion $\mathbf{q}_{\mathcal{W},t}^{\mathcal{R}}$.

D.5 WRIST POSE AND ROBOT FRAME TRANSFORMATION

For execution on a real robot, the recovered wrist trajectories must be expressed in the G1 base coordinate system rather than the camera frame. To achieve this, we first estimate the extrinsic transformation $\mathbf{T}_{\text{G1} \leftarrow w}$, parameterized as a rotation $\mathbf{R}_{\text{G1} \leftarrow w} \in \text{SO}(3)$ and a translation $\mathbf{t}_{\text{G1} \leftarrow w} \in \mathbb{R}^3$, between the reference camera world frame (camera in the first frame) and the G1 world frame, based on the known orientation of the G1 base and the viewing direction of the captured video. Applying this transformation maps each wrist position and orientation from the reference camera frame into the G1 base frame:

$$\mathbf{p}_{t,w}^{\text{G1}} = \mathbf{R}_{\text{G1} \leftarrow w} \mathbf{p}_{t,w}^{\text{w}} + \mathbf{t}_{\text{G1} \leftarrow w} \quad (9)$$

For Temporal Completion and Smoothing, we give the formulation as:

$$\hat{\mathbf{p}}_{\mathcal{W},t}^{\mathcal{R}} = \text{SG}(\tilde{\mathbf{p}}_{\mathcal{W},:}^{\mathcal{R}})_t, \quad \hat{\mathbf{q}}_{\mathcal{W},t}^{\mathcal{R}} = \frac{\text{SG}(\mathbf{q}_{\mathcal{W},:}^{\mathcal{R}})_t}{\|\text{SG}(\mathbf{q}_{\mathcal{W},:}^{\mathcal{R}})_t\|_2}. \quad (10)$$

The final wrist pose is $\hat{\mathbf{T}}_{\mathcal{R} \leftarrow \mathcal{W}, t} = (\hat{\mathbf{R}}(\hat{\mathbf{q}}_{\mathcal{W}, t}^{\mathcal{R}}), \hat{\mathbf{p}}_{\mathcal{W}, t}^{\mathcal{R}})$.

D.6 OTHER END-EFFECTORS RETARGETING: PARALLEL-JAW GRIPPERS.

We have already included how to generate five-finger robot hand motion in the main paper. Here we introduce how to generate motion for parallel-jaw grippers with our method. Retargeting a five-finger human hand directly to a two-finger gripper is inherently under-constrained, particularly for grasping motions. To address this, we employ *GraspNet* [29] to predict candidate grasp poses consistent with the observed scene geometry. Grasp execution is then triggered by heuristics that detect grasping intent from the human hand motion, ensuring reliable transfer of manipulation behaviors despite the reduced degrees of freedom.

E EXPERIMENTS: TASK SET AND TASKS DESCRIPTION

Task Name	Robot Setup	Task Description
Press Button	(1) Franka+Gripper	A robot arm is required to locate and press a button initialized at any arbitrary position within its reachable workspace.
Pick Objects	(1) Franka+Gripper	A robot arm is required to locate, grasp, and lift objects placed at arbitrary positions within its reachable workspace for manipulation.
Pick A into B	(1) Franka+Gripper	A robot arm is required to grasp object A and accurately place it into container B located at arbitrary positions in workspace.
Open Drawer	(1) Franka+Gripper	A robot arm is required to grasp the handle and pull open a drawer to a designated position.
Pick Objects (OOD Object)	(1) Franka+Gripper	A robot arm is required to pick unfamiliar, out-of-distribution objects randomly placed within its workspace.
Pick A into B (OOD Object)	(1) Franka+Gripper	A robot arm is required to place novel, out-of-distribution objects into container B within the workspace.
Pick Objects (OOD Scene)	(1) Franka+Gripper	A robot arm is required to pick objects in unseen, out-of-distribution scenes with new layouts or obstacles.
Pick A into B (OOD Scene)	(1) Franka+Gripper	A robot arm is required to place objects into container B under unseen, out-of-distribution scene configurations.
Pick Objects	(2) G1+DexHand	A dexterous robot hand is required to locate, grasp, and lift objects from arbitrary positions within workspace.
Open Box	(2) G1+DexHand	A dexterous robot hand is required to grasp and open a box lid placed in the workspace.
Open Door	(2) G1+DexHand	A dexterous robot hand is required to grasp and pull open a door handle to a designated position.
Wipe Table	(2) G1+DexHand	A dexterous robot hand is required to wipe the table surface clean using an object or cloth.
Press Elevator Button	(2) G1+DexHand	A dexterous robot hand is required to press an elevator button at any arbitrary position in the workspace.
Scoop Coffee Beans	(2) G1+DexHand	A dexterous robot hand is required to use a scoop to collect coffee beans and lift them successfully.
Sweep Tennis Ball into Bucket	(2) G1+DexHand	A dexterous robot hand is required to sweep a tennis ball across the surface into a target bucket.
Tear off Clear Tape	(2) G1+DexHand	A dexterous robot hand is required to peel and tear off a strip of transparent adhesive tape.

Table 3: Task Set and Tasks Description

F EXPERIMENTS: REAL-WORLD ROBOTS SET UP

Tasks	Robot	Control Frequency
Task Set1	Franka Panda Arm + Parallel-Jaw Gripper	15 Hz
Task Set2	Unitree G1 Arm + Inspire Hand (DH56DFX)	5 Hz

Table 4: Real-world robot experimental settings.

In Task Set 1, we used a Franka Emika Panda arm equipped with a parallel-jaw gripper for baseline grasping tasks.

In Task Set 2, we combined a Unitree G1 Humanoid with an Inspire dexterous hand. The two were mechanically connected via a flange, and we implemented synchronized arm-hand control at 5 Hz based on the Unitree teleoperation² framework. Furthermore, the joint angles predicted by our dex-retargeting module were remapped into the valid motor command ranges of the Inspire Hand to enable real-time execution.

G VIDEO RESULTS

We provide additional qualitative video results. For each video, eight frames are uniformly sampled for visualization. The first frame represents the input observation image, and the text caption below each sequence shows the task instruction (prior to rephrasing by the language model).

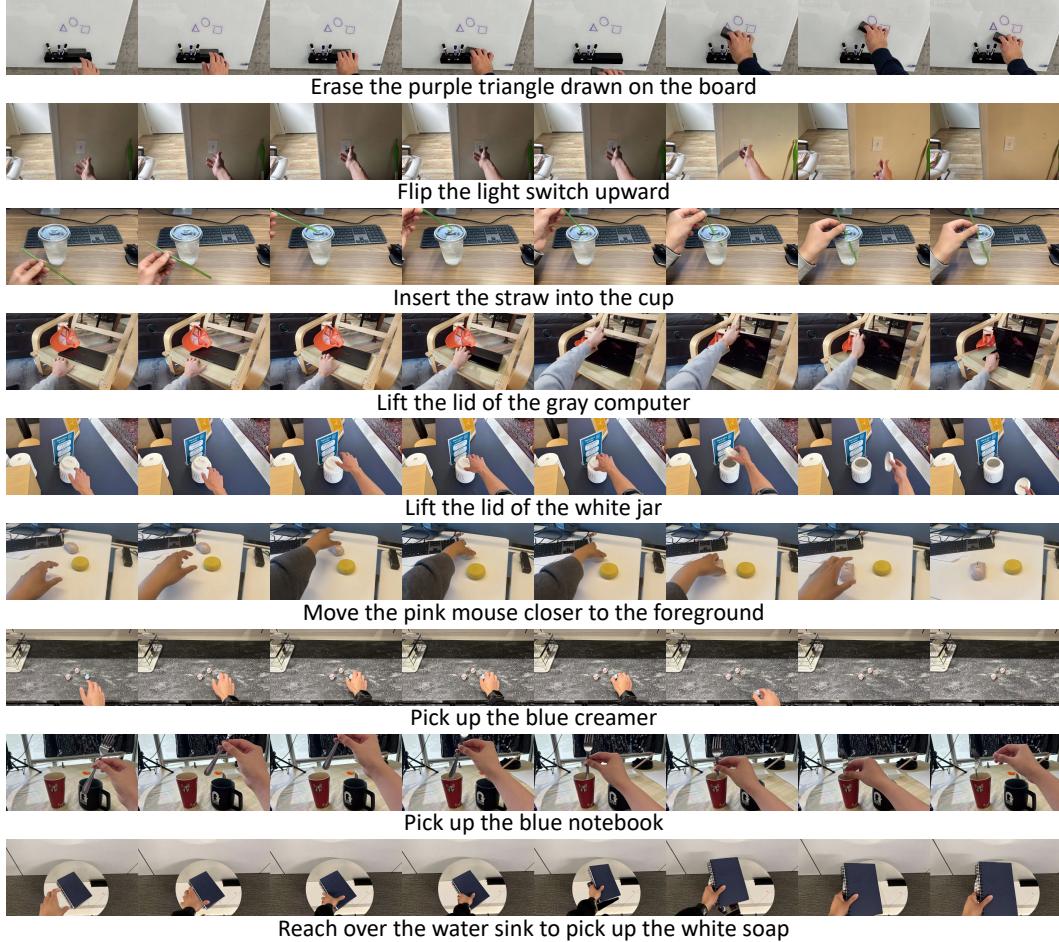


Figure 13: Visualization of generated video plans with human hand. Each row shows eight uniformly sampled frames from a generated video plan. The first frame depicts the input condition image that defines the scene. The caption below each sequence indicates the task instruction used to generate the video.

²https://github.com/unitreerobotics/xr_teleoperate

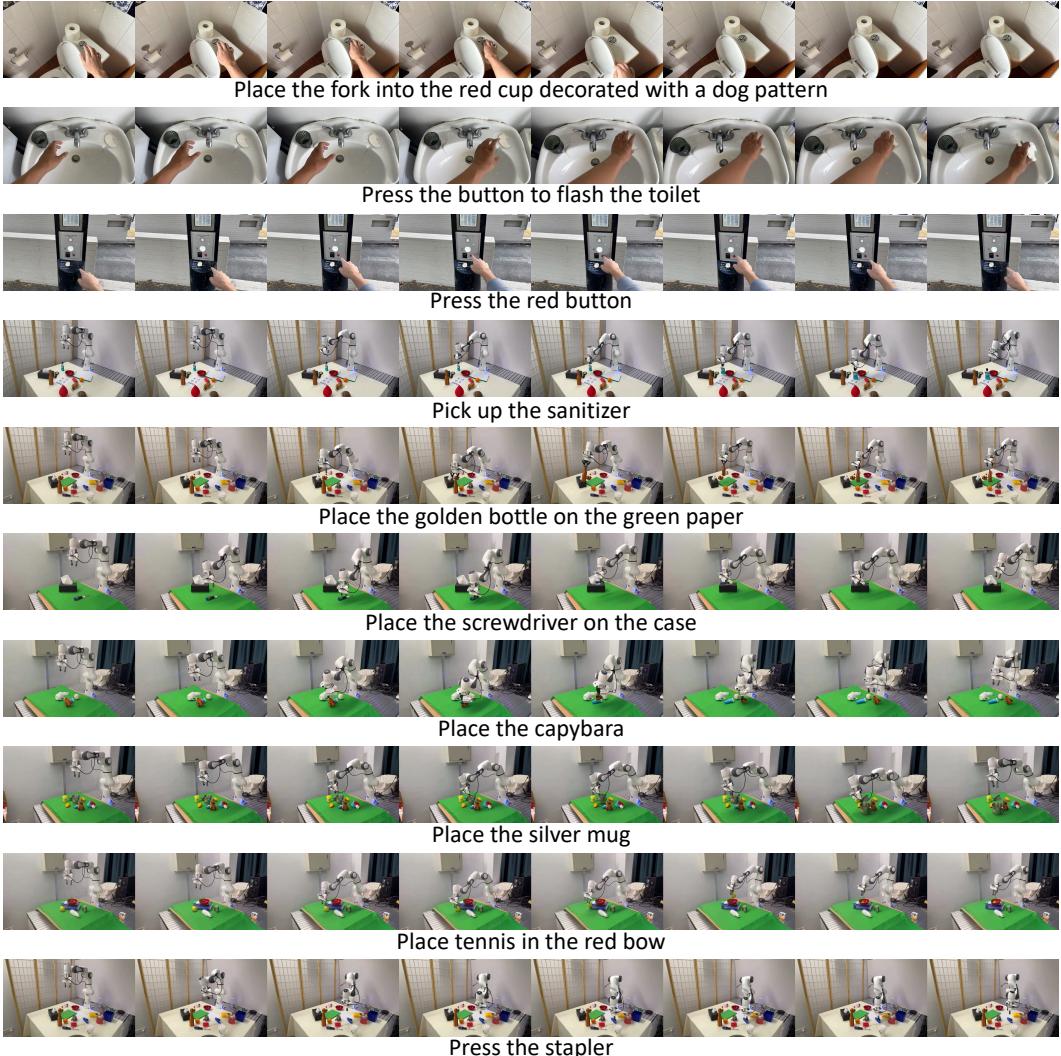


Figure 14: **Visualization of generated video plans with human hand and robot gripper.** Each row shows eight uniformly sampled frames from a generated video plan. The first frame depicts the input condition image that defines the scene. The caption below each sequence indicates the task instruction used to generate the video.



Figure 15: Visualization of generated videos and corresponding robot executions – A. The first row presents five uniformly sampled frames from the generated video plan conditioned on the input scene image. The second row illustrates the robot executing the same task in real world.

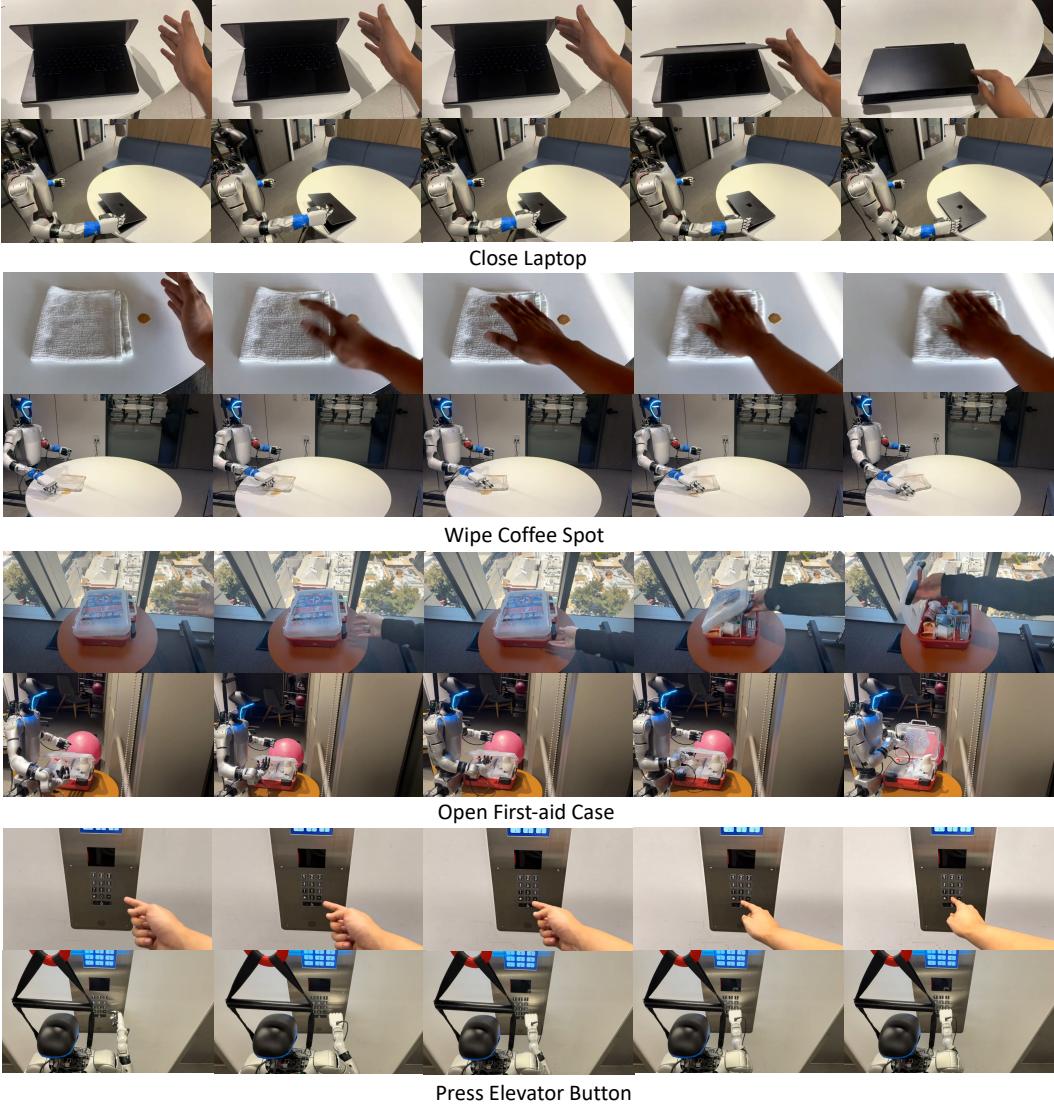


Figure 16: **Visualization of generated videos and corresponding robot executions – B.** The first row presents five uniformly sampled frames from the generated video plan conditioned on the input scene image. The second row illustrates the robot executing the same task in real world.