Gen2Act: Human Video Generation in Novel Scenarios enables Generalizable Robot Manipulation

Anonymous Author(s)
Affiliation
Address
email

Abstract: How can robot manipulation policies generalize to novel tasks involving unseen object types and new motions? In this paper, we provide a solution in terms of predicting motion information from web data through human video generation and conditioning a robot policy on the generated video. Instead of attempting to scale robot data collection which is expensive, we show how we can leverage video generation models trained on easily available web data, for enabling generalization. *Our approach Gen2Act casts language-conditioned manipulation as zero-shot human video generation followed by execution with a single policy conditioned on the generated video.* To train the policy, we use an order of magnitude less robot interaction data compared to what the video prediction model was trained on. *Gen2Act* doesn't require fine-tuning the video model at all and we directly use a pre-trained model for generating human videos. Our results on diverse real-world scenarios show how *Gen2Act* enables manipulating unseen object types and performing novel motions for tasks not present in the robot data.

Keywords: video generation, diverse manipulation

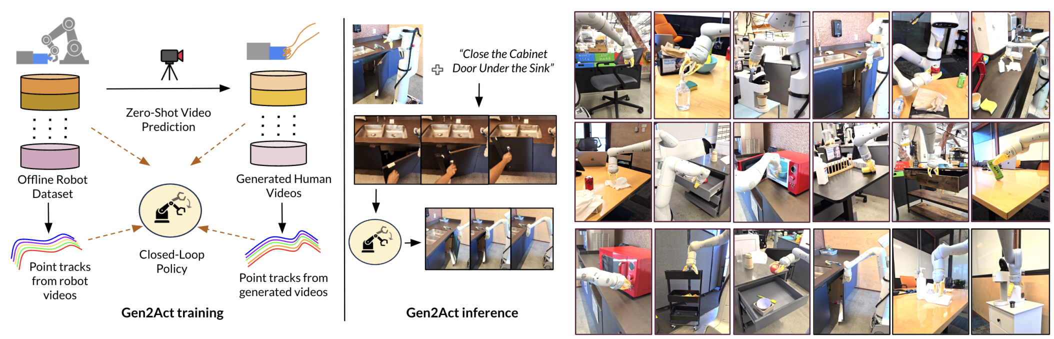

Figure 1: *Gen2Act* learns to generate a human video followed by robot policy execution conditioned on the generated video. This enables diverse real-world manipulation in unseen scenarios.

1 INTRODUCTION

To realize the vision of robot manipulators helping us in the humdrum everyday activities of messy living rooms, offices, and kitchens, it is crucial to develop robot policies capable of generalizing to novel tasks in unseen scenarios. In order to be practically useful, it is desirable to not require adapting the policy to new tasks through test-time optimizations and instead being able to directly execute it given a colloquial task specification such as language instructions. Further, such a policy should be able to tackle a broad array of everyday tasks like manipulating articulated objects, pouring, re-orienting objects, wiping tables without the need to collect robot interaction data for every

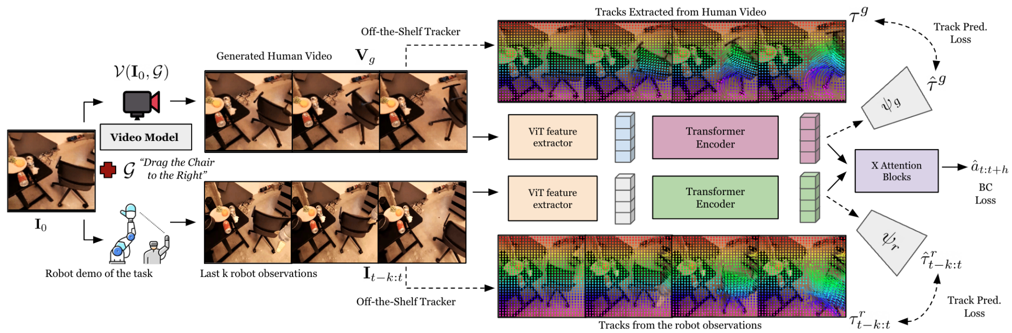

Figure 2: Architecture of the translation model of *Gen2Act* (closed-loop policy π_θ). Given an image of a scene $\mathbf{I}_0$ and a language-goal description of the task $\mathcal{G}$, we generate a human video $\mathbf{V}_g$ with a pre-trained video generation model $\mathcal{V}(\mathbf{I}_0, \mathcal{G})$. During training of the policy, we incorporate track prediction from the policy latents as an auxiliary loss in addition to a behavior cloning loss. Dotted pathways show training-specific computations. During inference, we do not require track prediction and only use the video model $\mathcal{V}$ in conjunction with the policy $\pi_\theta(\mathbf{I}_{t-k:t}, \mathbf{V}_g)$.

task unlike recent efforts on behavior cloning with robot datasets [1, 2, 3, 4]. This is because collecting large robot datasets that cover the diversity of everyday scenarios is extremely challenging and might be deemed impractical.

In order to mitigate issues with purely scaling robotic datasets, a line of recent works have sought to incorporate additional behavioral priors in representation learning by pre-training visual encoders with non-robotic datasets [5, 6, 7, 8, 9] and co-training policies with vision-language models [10, 11, 12]. Going beyond abstract representations, other works have learned attributes from web videos more directly informative of motion in the form of predicting goal images [13, 14, 15], hand-object mask plans [16], and embodiment-agnostic point tracks [17]. These approaches show promising signs of generalization to tasks unseen in the robot interaction datasets, but training such specific predictive models from web video data requires utilizing other intermediate models for providing ground-truths and thus are hard to scale up.

Our key insight for enabling generalization in manipulation is to cast motion prediction from web data in the very generic form of zero-shot video prediction. This lets us directly leverage advances in video generation models, by conditioning a robot policy on the generated video for new tasks that are unseen in the robot datasets. We posit that as video generation models get better due to large interest in generative AI [18, 19, 20] beyond robotics, an approach that relies on learning a policy conditioned on zero-shot video prediction can effectively scale and generalize to increasingly diverse real-world scenarios. For performing a manipulation task in a novel scene, a generated video conditioned on the language description of the task is particularly useful for conveying *what* needs to be done and in capturing motion-centric information of *how* to perform the task that can then be converted to robot actions through a learned policy. Compared to a generated video, a language description or a goal image alone only conveys what the task is.

We develop *Gen2Act* by instantiating language-conditioned manipulation as human video generation followed by generated human video to robot translation with a closed-loop policy (Fig. 1). We opt for generating human videos as opposed to directly generating robot videos since video generation models are often trained with human data on the web, and they are able to generate human videos zero-shot given a new scene. We then train a translation model that needs some offline robot demonstrations and corresponding generated human videos. We generate these corresponding human videos offline with an off-the-shelf model [20] by conditioning on the first frame of each trajectory (the first frame doesn't have the robot in the scene) and the language description of the task. We instantiate this translation model as a closed loop policy that is conditioned on the history of robot observations in addition to the generated human video so that it can take advantage of the visual cues in the scene and adjust its behavior reactively.

In order to capture motion information beyond that implicitly provided by visual features from the generated video, we extract point tracks from the generated human video and the video of robot observations (through an off-the-shelf tracker [21]) and optimize a track prediction auxiliary loss during training. The aim of this loss function is to ensure that the latent tokens of the closed-loop policy are informative of the motion of points in the scene. We train the policy to optimize the typical behavior cloning loss for action prediction combined with this track prediction loss. For deployment, give a language description of a task to be performed, we generate a human video and run the policy conditioned on this video.

The diverse real-world manipulation results of *Gen2Act* (featured in Fig. 1) demonstrate the broad generalization capabilities enabled by learning to infer motion cues from web video data through zero-shot video generation combined with motion extraction through point track prediction for solving novel manipulation tasks in unseen scenarios. For generalization to novel object types and novel motion types unseen in the robot interaction training data, we show that *Gen2Act* achieves on average $\sim 30\%$ higher absolute success rate over the most competitive baseline. Further, we demonstrate how *Gen2Act* can be chained in sequence for performing long-horizon activities like "making coffee" consisting of several intermediate tasks.

2 Related Works

We discuss prior works in imitation learning with visual observations, learning representations from non-robotic datasets, and approaches for conditional behavior cloning.

Visual Imitation. Visual imitation is a scalable approach for robotic manipulation [22, 23, 24] and end-to-end policy learning more broadly [25, 26]. While early works in multi-task imitation learning collected limited real-world data [27, 28], more recent approaches [29, 1, 30] collect much larger datasets. In fact, recent works that have attempted to directly scale this for training large models have required years of expensive data collection [1, 10, 2] and have still been restricted to limited generalization especially with respect to novel object types and novel motions in unseen scenarios.

Visual Representations for Manipulation. To enable generalization, many recent works propose using pre-trained visual representations trained primarily on non-robot datasets [31, 32], for learning manipulation policies [5, 8, 6, 33, 6, 34, 7, 9, 35, 36]. However, they are primarily limited to learning task-specific policies [5, 8, 37, 38] as they rely on access to a lot of in-domain robot interaction data. Apart from training visual encoders, a line of works augment existing robot datasets with semantic variations using generative models [39, 40, 41, 2, 42]. While this enables policies to generalize to unseen scenes and become robust to distractors, generalization to unseen object types and motion types still remains a challenge.

Conditional Behavior Cloning. Some prior works train robotic policies conditioned on human videos but require paired in-domain human-robot data [43, 44, 45, 46, 47, 48] and are not capable of leveraging web data. Others use curated data of human videos to leverage human hand motion information [49, 50] for learning task-specific policies (instead of a single model across generic tasks). Towards learning structure more directly related to manipulation from web videos, some works try to predict visual affordances in the form of where to interact in an image, and local information of how to interact [51, 52, 53, 54, 55]. While these could serve as good initializations for a robotic policy, they are not sufficient on their own for accomplishing tasks, and so are typically used in conjunction with online learning, requiring several hours of deployment-time training and robot data [56, 53, 13]. Others learn to predict motion from web data more directly in the form of masks of hand and objects in the scene [16] and tracks of how arbitrary points in the scene should move [17], for conditional behavior cloning. However, training such predictive models from web videos requires reliance on intermediate models for providing ground-truth information and are thus hard to scale up broadly.

3 APPROACH

We develop a language-conditioned robot manipulation system, *Gen2Act* that generalizes to novel tasks in unseen scenarios. To achieve this, we adopt a factorized approach: 1) Given a scene and a task description, using an existing video prediction model generate a video of a human solving the task, 2) Conditioned on the generated human video infer robot actions through a learned human-to-robot translation model that can take advantage of the motion cues in the generated video. We show that this factorized strategy is scalable in leveraging web-scale motion understanding inherent in large video models, for synthesizing *how* the manipulation should happen for a novel task, and utilizing orders of magnitude less robot interaction data for the much simpler task of translation from a generated human video to *what* actions the robot should execute.

3.1 Overview and Setup

Given a scene specified by an image $\mathbf{I}_0$ and a goal $\mathcal{G}$ describing in text the task to be performed, we want a robot manipulation system to execute actions $\mathbf{a}_{1:H}$ for solving the task. To achieve this in unseen scenarios, we learn motion predictive information from web video data in the form of a video prediction model $\mathcal{V}(\mathbf{I}_0, \mathcal{G})$ that zero-shot generates a human video of the task, $\mathbf{V}_g$. In order to translate this generated video to robot actions, we train a closed-loop policy $\pi_\theta(\mathbf{I}_{t-k:t}, \mathbf{V}_g)$ conditioned on the video and the last k robot observations, through behavior cloning on a small robot interaction dataset $\mathcal{D}_r$. In order to implicitly encode motion information from $\mathbf{V}_g$ in the policy π_θ, we extract point tracks from both $\mathbf{V}_g$ and $\mathbf{I}_{t-k:t}$, respectively τ_g and τ_r, and incorporate track prediction as an auxiliary loss $\mathcal{L}_\tau$ during training. Fig. 2 shows an overview of this setup.

3.2 Human Video Generation

We use an existing video generation model for the task of text+image conditioned video generation. We find that current video generation models are good at generating human videos zero-shot without requiring any fine-tuning or adaptation (some examples in Fig. 3). Instead of trying to generate robot videos as done by some prior works [57, 58], we focus on just human video generation because current video generation models cannot generate robot videos zero-shot and require robot-specific fine-tuning data for achieving this. Such fine-tuning often subtracts the benefits of generalization to novel scenes that is inherent in video generation models trained on web-scale data.

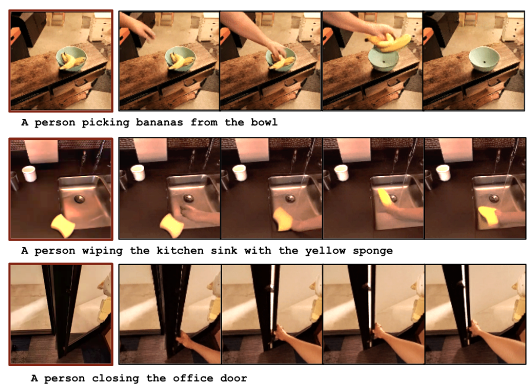

Figure 3: Visualization of zero-shot video generation for different tasks. The blue frame and the language description are input to the video generation model of *Gen2Act* and the black frames show sub-sampled frames of the generated video. These results demonstrate the applicability of off-the-shelf video generation models for image+text conditioned video generation that preserves the scene and performs the desired manipulation task.

For training, given an offline dataset of robot trajectories $\mathcal{D}_r$ along with language task instructions $\mathcal{G}$, we create a corresponding generated human video dataset $\mathcal{D}_g$ by generating videos conditioned on the first frame of the robot trajectories and the language instruction. This procedure of generating paired datasets $\{\mathcal{D}_r, \mathcal{D}_g\}$ is fully automatic and does not require manually collecting human videos as done by prior works [59, 46]. We do not require the generated human videos to have any particular structure apart from looking visually realistic, manipulating the relevant objects plausibly, and having minimal camera motion. As seen in the qualitative results in Fig. 3, all of this is achieved zero-shot with a pre-trained video model.

During evaluation, we move the robot to a new scene $\mathbf{I}_0$, specify a task to be performed in language $\mathcal{G}$, and then generate a human video $\mathbf{V}_g = \mathcal{V}(\mathbf{I}_0, \mathcal{G})$ that is fed into the human-to-robot translation policy, described in Section 3.3. Our approach is not tied to a specific video generative model and as video models become better, this stage of our approach will likely scale upwards. We expect the overall approach to generalize as well since the translation model is tasked with a simpler job of inferring motion cues from the generated human video in novel scenarios, and implicitly converting that to robot actions. As we show through results in Section 3.3 only a small amount of diverse robot trajectories (~ 400) combined with existing offline datasets is enough to train a robust translation model.

3.3 Generated Human Video to Robot Action Translation

We instantiate generated human video to robot action translation as a closed loop policy π_θ. Given a new scene and a task description, the generated human video provides motion cues for how the manipulation should happen in the scene, and the role of the policy is to leverage relevant information from the generated video, combined with observations in the robot's frame, for interacting in the scene. Instead of attempting to explicitly extract waypoints from the generated video based on heuristics, we adopt a more end-to-end approach that relies on general visual features of the video, and general point tracks extracted from the video. This implicit conditioning on the generated video is helpful in mitigating potential artifacts in the generation and in making the approach more robust to mismatch in the video and the robot's embodiment. Note that we perform human video generation and ground-truth track extraction completely offline for training.

Visual Feature Extraction. For each frame in the generated human video $\mathbf{V}_g$ and the robot video $\mathbf{I}_{t-k:k}$, we first extract features, i_g and i_r through a ViT encoder χ. The number of video tokens extracted this way is very large and they are temporally uncorrelated, so we have Transformer encoders Φ_g and Φ_r that process the respective video tokens through gated Cross-Attention Layers based on a Perceiver-Resampler architecture [60] and output a fixed number $N = 64$ of tokens. These tokens respectively are $z_g = \Phi_g(i_g)$ and $z_r = \Phi_r(i_r)$.

In addition to visual features from the generated video, we encode explicit motion information in the human-to-robot translation policy through point track prediction.

Point Track Prediction. We run an off-the-shelf tracking model [61, 21] on the generated video $\mathbf{V}_g$ to obtain tracks τ_g of a random set of points in the first frame P^0. In order to ensure that the latent embeddings from the generated video z_g can distill motion information in the video, we set up a track prediction task conditioned on the video tokens. For this, we define a track prediction transformer $\psi_g(P^0, i_g^0, z_g)$ to predict tracks $\hat{\tau_g}$ and define an auxiliary loss $||\tau_g - \hat{\tau_g}||_2$ to update tokens g_e.

Similarly, for the current robot video $\mathbf{I}_r^{t-k:k}$, we set up a similar track prediction auxiliary loss. We run the ground-truth track prediction once over the entire robot observation sequence (again with random points in the first frame P_0), but during training, the policy is input a chunk of length k in one pass. So here, the track prediction transformer $\psi_r(P^{t-k}, i_r^{t-k}., z_r^{t-k:t})$ is conditioned on the points in the beginning of the chunk P_{t-k}, the image features at that time-step i^{t-k} and the observation tokens for the chunk z_r.

BC Loss. For ease of prediction, we discretize the action space such that each dimension has 256 bins. We optimize a Behavior Cloning (BC) objective by minimizing error between the predicted actions $\hat{a}_{t:t+h}$ and the ground-truth $a_{t:t+h}$ through a cross-entropy loss.

In *Gen2Act*, we incorporate track prediction as an auxiliary loss during training combined with the BC loss and the track prediction transformer is not used at test-time. This is helpful in reducing test-time computations for efficient deployment.

3.4 Deployment

For deploying *Gen2Act* to solve a manipulation task, we first generate a human video conditioned on the language description of the task and the image of the scene. We then roll out the generated video conditioned closed-loop policy. For chaining *Gen2Act* to perform long-horizon activities consisting of several tasks, we first use an off-the-shelf LLM (e.g. Gemini) to obtain language descriptions of the different tasks. We chain *Gen2Act* for the task sequence by using the last image of the previous policy rollout as the first frame for generating a human video of the subsequent task. We do this

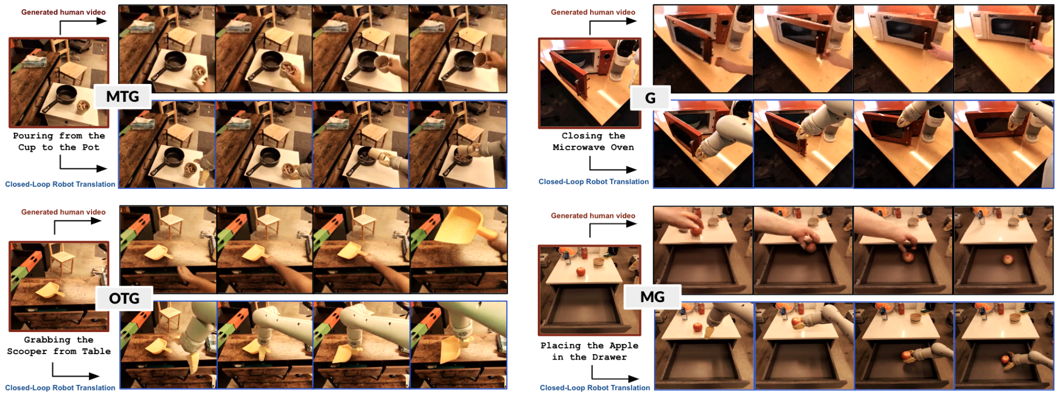

Figure 4: Visualization of the closed-loop policy rollouts (bottom row) conditioned on the generated human videos (top row) for four tasks. The red frame and the language description are input to the video generation model of *Gen2Act* . The black frames show sub-sampled frames of the generated video, and the blue frames show robot executions conditioned on the generated video.

chaining in sequence as opposed to generating all the videos from the first image because the final state of the objects in the scene might be different after the robot execution of an intermediate task.

4 EXPERIMENTS

We perform experiments in diverse kitchen, office, and lab scenes, across a wide array of manipulation tasks. Through these experiments we aim to answer the following questions:

- Is *Gen2Act* able to generate plausible human videos of manipulation in diverse everyday scenes?
- How does *Gen2Act* perform in terms of varying levels of generalization with new scenes, objects, and motions?
- Can *Gen2Act* enable long-horizon manipulation through chaining of the video generation and video-conditioned policy execution?
- Can the performance of *Gen2Act* for new tasks be improved by co-training with a small amount of additional diverse human tele-operated demonstrations?

4.1 Details of the Evaluation Setup

Following prior works in language/goal-conditioned policy learning, we quantify success in terms of whether the executed robot trajectory solves the task specified in the instruction, and define success rate over different rollouts for the same task description. We categorize evaluations with respect to different levels of generalization by following the terminology of prior works [17, 1]: Mild Generalization (**MG**): unseen configurations of seen object instances in seen scenes; organic scene variations like lighting and background changes. Standard Generalization (**G**): unseen object instances in seen/unseen scenes. Object-Type Generalization (**OTG**): completely unseen object types, in unseen scenes. Motion-Type Generalization (**MTG**): completely unseen motion types, in unseen scenes

Here, seen vs. unseen is defined with respect to the robot interaction data, and the assumption is that the video generation model has seen diverse web data including things that are unseen in the robot data.

4.2 Dataset and hardware details

For video generation, we use an existing video model, VideoPoet [20] For obtaining tracks on the generated human video and the robot demo, we use an off-the-shelf tracking approach [61, 21]. For robot experiments, we use a mobile manipulator with compliant two finger-grippers, and operate

Table 1: Comparison of success rates for *Gen2Act* with different baselines and an ablated variant for the different levels of generalization as defined in Section 4.1

	Mild (MG)	Standard (G)	Obj. Type (OTG)	Motion. Type (MTG)	Avg.
RT1	68	18	0	0	22
RT1-GC	75	24	5	0	26
Vid2Robot	83	38	25	0	37
Gen2Act (w/o track)	83	58	50	5	49
Gen2Act	83	67	58	30	60

this robot for policy deployment through end-effector control. The arm is attached to the body of the robot on the right. We manually move the robot around across offices, kitchens, and labs and ask it to manipulate different objects in these scenes. We operate the robot for manipulation at a frequency of 3Hz. Before each task, we reset the robot arm to a fixed pre-defined reset position such that the scene is not occluded through the robot's camera.

4.3 Baselines and Comparisons

We perform comparisons with baselines and ablations with variants of *Gen2Act*. In particular, we compare with a language-conditioned policy baseline (*RT1*) [1] trained on the same robot data as *Gen2Act*. We also compare with a video-conditioned policy baseline trained on paired real human and robot videos (*Vid2Robot*) [46], a goal-image conditioned policy baseline trained with the same real and generated videos of *Gen2Act* but by conditioning on just the last video frames (i.e. goal image) of the generated human videos (*RT1-GC*). Finally, we consider an ablated variant of *Gen2Act* without the track prediction loss.

4.4 Analysis of Human Video Generations

Fig. 3 shows qualitative results for human video generation in diverse scenarios. We can see that the generated videos correspond to plausibly manipulating the scene in the initial image as described by the text instruction. We can see that the respective object in the scene is manipulated while preserving the background and without introducing camera movements and artifacts in the generations. This is exciting because these generations are zero-shot in novel scenarios and can be directly used in a robot's context to imagine how an unseen object in an unseen scene should be manipulated by a human.

4.5 Generalization of *Gen2Act* to scenes, objects, motions

In this section we compare performance of *Gen2Act* with baselines and ablated variants for different levels of generalization. Table 1 shows success rates for tasks averaged across different levels of generalization. We observe that for higher levels of generalization, *Gen2Act* achieves much higher success rates indicating that human video generation combined with explicitly extracting motion information from track prediction is helpful in unseen tasks.

4.6 Chaining *Gen2Act* for long-horizon manipulation

We now analyze the feasibility of *Gen2Act* for solving a sequence of manipulation tasks through chaining. Table 3 shows results for long-horizon activities like "Making Coffee" that consist of multiple tasks to be performed in sequence. We obtain this sequence of tasks through Gemini [62], and for each task, condition the video generation on the last image of the scene from the previous execution and execute the policy for the current task conditioned on the generated human video. We repeat this in sequence for all the stages, and report success rates for successful completion upto each stage over 5 trials. Fig. 5 visually illustrates single-take rollouts from four such long-horizon activities.

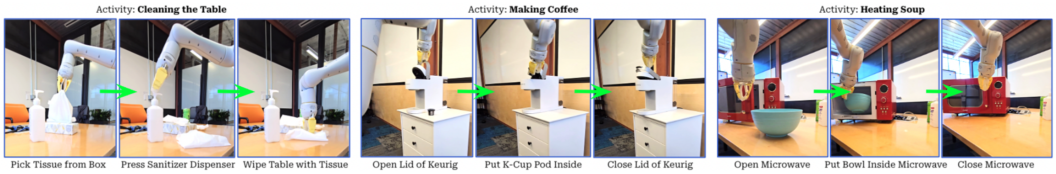

Figure 5: Robot executions for a sequence of tasks. The last frame of the previous execution serves as the conditioning frame for next stage video generation.

Table 2: Analysis of co-training with an additional dataset of diverse tele-operated robot demonstrations ($\sim$ 400 trajectories).

Co-Training	**Mild (MG)**	**Standard (G)**	**Obj. Type (OTG)**	**Motion. Type (MTG)**	**Avg.**
Gen2Act (w/o co-train)	83	67	58	30	60
Gen2Act (w/ co-train)	85	75	62	35	64

4.7 Co-Training with additional teleop demonstrations

The offline dataset we used for experiments in the previous section had limited coverage over scenes and types of tasks thereby allowing less than 60% success rate of *Gen2Act* for higher levels of generalization (OTG and MTG in Table 1). In this section, we perform experiments to understand if adding a small amount of additional *diverse* tele-operated trajectories, for co-training with the existing offline dataset, can help improve generalization. We keep the video generation model fixed as usual. From the results in Table 2 we see improved performance of *Gen2Act* with such co-training. This is exciting because it suggests that with only a small amount of diverse demonstrations, the translation model of *Gen2Act* can be improved to better condition on the generated videos for higher levels of generalization where robot data support is limited.

4.8 Analysis of Failures

Here we discuss the type of failures exhibited by *Gen2Act*. We observe that for MG and to some extent in G, inaccuracies in video generation are less correlated with failures of the policy. While, for the higher levels of generalization, object type (OTG) and motion type (MTG), if video generation yields implausible videos, then the policy doesn't succeed in performing the tasks. This is also evidence that the policy of *Gen2Act* is using the generated human video for inferring motion cues while completing a task, and as such when video generation is incorrect in scenarios where robot data support is limited (e.g. in OTG and MTG), the policy fails.

5 Discussion and Conclusion

Summary. In this work, we developed a framework for learning generalizable robot manipulation by combining zero-shot human video generation from web data with limited robot demonstrations. Broadly, our work is indicative of how motion predictive models trained on non-robotic datasets like web videos can be used to used to enable generalization of manipulation policies to unseen scenarios, without requiring collection of robot data for every task.

Limitations. Our work focused on zero-shot human video generation combined with point track prediction on the videos as a way for providing motion cues to a robot manipulation system for interacting with unseen objects and performing novel tasks. As such, the capabilities of our system are limited by the current capabilities of video generation models, like inability to generate realistic hands and thereby limited ability to perform very dexterous tasks.

Future Work. It would be an interesting direction of future work to explore recovering more dense motion information from the generated videos beyond point tracks, like object meshes for addressing some of the limitations. Another important direction would be to enable reliable long-horizon manipulation by augmenting chaining with learning recovery policies for intermediate failures.

References

[1] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

[2] H. Bharadhwaj, J. Vakil, M. Sharma, A. Gupta, S. Tulsiani, and V. Kumar. Roboagent: Generalization and efficiency in robot manipulation via semantic augmentations and action chunking. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[3] E. Jang, A. Irpan, M. Khansari, D. Kappler, F. Ebert, C. Lynch, S. Levine, and C. Finn. Bc-z: Zero-shot task generalization with robotic imitation learning. In *Conference on Robot Learning*, pages 991–1002. PMLR, 2022.

[4] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany, M. K. Srirama, L. Y. Chen, K. Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

[5] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta. R3m: A universal visual representation for robot manipulation. *arXiv preprint arXiv:2203.12601*, 2022.

[6] A. Majumdar, K. Yadav, S. Arnaud, Y. J. Ma, C. Chen, S. Silwal, A. Jain, V.-P. Berges, P. Abbeel, J. Malik, et al. Where are we in the search for an artificial visual cortex for embodied intelligence? *arXiv preprint arXiv:2303.18240*, 2023.

[7] T. Xiao, I. Radosavovic, T. Darrell, and J. Malik. Masked visual pre-training for motor control. *arXiv preprint arXiv:2203.06173*, 2022.

[8] S. Parisi, A. Rajeswaran, S. Purushwalkam, and A. Gupta. The unsurprising effectiveness of pre-trained vision models for control. *arXiv preprint arXiv:2203.03580*, 2022.

[9] S. Karamcheti, S. Nair, A. S. Chen, T. Kollar, C. Finn, D. Sadigh, and P. Liang. Language-driven representation learning for robotics. *arXiv preprint arXiv:2302.12766*, 2023.

[10] B. Zitkovich, T. Yu, S. Xu, P. Xu, T. Xiao, F. Xia, J. Wu, P. Wohlhart, S. Welker, A. Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pages 2165–2183. PMLR, 2023.

[11] N. D. Palo and E. Johns. Dinobot: Robot manipulation via retrieval and alignment with vision foundation models. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[12] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

[13] K. Black, M. Nakamoto, P. Atreya, H. Walke, C. Finn, A. Kumar, and S. Levine. Zero-shot robotic manipulation with pretrained image-editing diffusion models. *arXiv preprint arXiv:2310.10639*, 2023.

[14] H. Bharadhwaj, A. Gupta, and S. Tulsiani. Visual affordance prediction for guiding robot exploration. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3029–3036. IEEE, 2023.

[15] I. Kapelyukh, V. Vosylius, and E. Johns. Dall-e-bot: Introducing web-scale diffusion models to robotics. *IEEE Robotics and Automation Letters*, 8(7):3956–3963, 2023.

[16] H. Bharadhwaj, A. Gupta, V. Kumar, and S. Tulsiani. Towards generalizable zero-shot manipulation via translating human interaction plans. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[17] H. Bharadhwaj, R. Mottaghi, A. Gupta, and S. Tulsiani. Track2act: Predicting point tracks from internet videos enables diverse zero-shot robot manipulation. *arXiv preprint arXiv:2405.01527*, 2024.

[18] R. Girdhar, M. Singh, A. Brown, Q. Duval, S. Azadi, S. S. Rambhatla, A. Shah, X. Yin, D. Parikh, and I. Misra. Emu video: Factorizing text-to-video generation by explicit image conditioning. *arXiv preprint arXiv:2311.10709*, 2023.

[19] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. Denton, S. K. S. Ghasemipour, B. K. Ayan, S. S. Mahdavi, R. G. Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022.

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

[21] C. Doersch, Y. Yang, D. Gokay, P. Luc, S. Koppula, A. Gupta, J. Heyward, R. Goroshin, J. Carreira, and A. Zisserman. Bootstap: Bootstrapped training for tracking-any-point. *arXiv preprint arXiv:2402.00847*, 2024.

[22] C. Finn, T. Yu, T. Zhang, P. Abbeel, and S. Levine. One-shot visual imitation learning via meta-learning. In *Conference on robot learning*, pages 357–368. PMLR, 2017.

[23] S. Young, D. Gandhi, S. Tulsiani, A. Gupta, P. Abbeel, and L. Pinto. Visual imitation made easy. In *Conference on Robot Learning (CoRL)*, 2020.

[24] A. Mandlekar, Y. Zhu, A. Garg, J. Booher, M. Spero, A. Tung, J. Gao, J. Emmons, A. Gupta, E. Orbay, et al. Roboturk: A crowdsourcing platform for robotic skill learning through imitation. In *Conference on Robot Learning*, pages 879–893. PMLR, 2018.

[25] S. Ross, N. Melik-Barkhudarov, K. S. Shankar, A. Wendel, D. Dey, J. A. Bagnell, and M. Hebert. Learning monocular reactive uav control in cluttered natural environments. In *2013 IEEE international conference on robotics and automation*, pages 1765–1772. IEEE, 2013.

[26] Z. Chen and X. Huang. End-to-end learning for lane keeping of self-driving cars. In *2017 IEEE intelligent vehicles symposium (IV)*, pages 1856–1860. IEEE, 2017.

[27] A. Mandlekar, Y. Zhu, A. Garg, J. Booher, M. Spero, A. Tung, J. Gao, J. Emmons, A. Gupta, E. Orbay, et al. Roboturk: A crowdsourcing platform for robotic skill learning through imitation. In *Conference on Robot Learning*, pages 879–893. PMLR, 2018.

[28] E. Jang, A. Irpan, M. Khansari, D. Kappler, F. Ebert, C. Lynch, S. Levine, and C. Finn. Bc-z: Zero-shot task generalization with robotic imitation learning. In *Conference on Robot Learning*, pages 991–1002. PMLR, 2022.

[29] H. R. Walke, K. Black, T. Z. Zhao, Q. Vuong, C. Zheng, P. Hansen-Estruch, A. W. He, V. Myers, M. J. Kim, M. Du, et al. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning*, pages 1723–1736. PMLR, 2023.

[30] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke, et al. Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on robot learning*, pages 651–673. PMLR, 2018.

[31] K. Grauman, A. Westbury, E. Byrne, Z. Chavis, A. Furnari, R. Girdhar, J. Hamburger, H. Jiang, M. Liu, X. 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*, pages 18995–19012, 2022.

[32] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[33] Y. J. Ma, S. Sodhani, D. Jayaraman, O. Bastani, V. Kumar, and A. Zhang. Vip: Towards universal visual reward and representation via value-implicit pre-training. *arXiv preprint arXiv:2210.00030*, 2022.

[34] S. Parisi, A. Rajeswaran, S. Purushwalkam, and A. Gupta. The unsurprising effectiveness of pre-trained vision models for control. *arXiv preprint arXiv:2203.03580*, 2022.

[35] H. Wu, Y. Jing, C. Cheang, G. Chen, J. Xu, X. Li, M. Liu, H. Li, and T. Kong. Unleashing large-scale video generative pre-training for visual robot manipulation. *arXiv preprint arXiv:2312.13139*, 2023.

[36] S. Yang, J. Walker, J. Parker-Holder, Y. Du, J. Bruce, A. Barreto, P. Abbeel, and D. Schuurmans. Video as the new language for real-world decision making. *arXiv preprint arXiv:2402.17139*, 2024.

[37] M. Sharma, C. Fantacci, Y. Zhou, S. Koppula, N. Heess, J. Scholz, and Y. Aytar. Lossless adaptation of pretrained vision models for robotic manipulation. *arXiv preprint arXiv:2304.06600*, 2023.

[38] N. Hansen, Z. Yuan, Y. Ze, T. Mu, A. Rajeswaran, H. Su, H. Xu, and X. Wang. On pre-training for visuo-motor control: Revisiting a learning-from-scratch baseline. *arXiv preprint arXiv:2212.05749*, 2022.

[39] Z. Mandi, H. Bharadhwaj, V. Moens, S. Song, A. Rajeswaran, and V. Kumar. Cacti: A framework for scalable multi-task multi-scene visual imitation learning. *arXiv preprint arXiv:2212.05711*, 2022.

[40] Z. Chen, S. Kiami, A. Gupta, and V. Kumar. Genaug: Retargeting behaviors to unseen situations via generative augmentation. *arXiv preprint arXiv:2302.06671*, 2023.

[41] T. Yu, T. Xiao, A. Stone, J. Tompson, A. Brohan, S. Wang, J. Singh, C. Tan, J. Peralta, B. Ichter, et al. Scaling robot learning with semantically imagined experience. *arXiv preprint arXiv:2302.11550*, 2023.

[42] Z. Chen, Z. Mandi, H. Bharadhwaj, M. Sharma, S. Song, A. Gupta, and V. Kumar. Semantically controllable augmentations for generalizable robot learning. *arXiv preprint arXiv:2409.00951*, 2024.

[43] C. Wang, L. Fan, J. Sun, R. Zhang, L. Fei-Fei, D. Xu, Y. Zhu, and A. Anandkumar. Mimicplay: Long-horizon imitation learning by watching human play. *arXiv preprint arXiv:2302.12422*, 2023.

[44] L. Smith, N. Dhawan, M. Zhang, P. Abbeel, and S. Levine. Avid: Learning multi-stage tasks via pixel-level translation of human videos. *arXiv*, 2019.

[45] H. Xiong, Q. Li, Y.-C. Chen, H. Bharadhwaj, S. Sinha, and A. Garg. Learning by watching: Physical imitation of manipulation skills from human videos. *arXiv*, 2021.

[46] V. Jain, M. Attarian, N. J. Joshi, A. Wahid, D. Driess, Q. Vuong, P. R. Sanketi, P. Sermanet, S. Welker, C. Chan, et al. Vid2robot: End-to-end video-conditioned policy learning with cross-attention transformers. *arXiv preprint arXiv:2403.12943*, 2024.

[47] C. Wen, X. Lin, J. So, K. Chen, Q. Dou, Y. Gao, and P. Abbeel. Any-point trajectory modeling for policy learning. *arXiv preprint arXiv:2401.00025*, 2023.

[48] J. Gu, S. Kirmani, P. Wohlhart, Y. Lu, M. G. Arenas, K. Rao, W. Yu, C. Fu, K. Gopalakrishnan, Z. Xu, et al. Rt-trajectory: Robotic task generalization via hindsight trajectory sketches. *arXiv preprint arXiv:2311.01977*, 2023.

[49] Y. Qin, Y.-H. Wu, S. Liu, H. Jiang, R. Yang, Y. Fu, and X. Wang. Dexmv: Imitation learning for dexterous manipulation from human videos. *arXiv preprint arXiv:2108.05877*, 2021.

[50] K. Shaw, S. Bahl, and D. Pathak. Videodex: Learning dexterity from internet videos. In *6th Annual Conference on Robot Learning*.

[51] K. Mo, L. J. Guibas, M. Mukadam, A. Gupta, and S. Tulsiani. Where2act: From pixels to actions for articulated 3d objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6813–6823, 2021.

[52] M. Goyal, S. Modi, R. Goyal, and S. Gupta. Human hands as probes for interactive object understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3293–3303, 2022.

[53] S. Bahl, R. Mendonca, L. Chen, U. Jain, and D. Pathak. Affordances from human videos as a versatile representation for robotics. In *CVPR*, 2023.

[54] S. Liu, S. Tripathi, S. Majumdar, and X. Wang. Joint hand motion and interaction hotspots prediction from egocentric videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3282–3292, 2022.

[55] C. Yuan, C. Wen, T. Zhang, and Y. Gao. General flow as foundation affordance for scalable robot learning. *arXiv preprint arXiv:2401.11439*, 2024.

[56] S. Bahl, A. Gupta, and D. Pathak. Human-to-robot imitation in the wild. *RSS*, 2022.

[57] Y. Du, S. Yang, B. Dai, H. Dai, O. Nachum, J. Tenenbaum, D. Schuurmans, and P. Abbeel. Learning universal policies via text-guided video generation. *Advances in Neural Information Processing Systems*, 36, 2024.

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

[59] G. Papagiannis, N. Di Palo, P. Vitiello, and E. Johns. R+ x: Retrieval and execution from everyday human videos. *arXiv preprint arXiv:2407.12957*, 2024.

[60] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

[61] C. Doersch, A. Gupta, L. Markeeva, A. Recasens, L. Smaira, Y. Aytar, J. Carreira, A. Zisserman, and Y. Yang. Tap-vid: A benchmark for tracking any point in a video. *Advances in Neural Information Processing Systems*, 35:13610–13626, 2022.

[62] G. Team, R. Anil, S. Borgeaud, Y. Wu, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[63] N. Karaev, I. Rocco, B. Graham, N. Neverova, A. Vedaldi, and C. Rupprecht. Cotracker: It is better to track together. *arXiv preprint arXiv:2307.07635*, 2023.

[64] G. Le Moing, J. Ponce, and C. Schmid. Dense optical tracking: connecting the dots. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19187–19197, 2024.

Appendix

Here we provide additional details on the method and experiments of *Gen2Act*.

5.1 Human Video Generation

We use a pre-trained VideoPoet model [20] directly without any adaptation or fine-tuning. The input to the model for video generation is a language description of a task (the prompt) and a square-shaped image. By virtue of being trained on diverse large-scale video datasets ($> 270M$ videos) we find that this model generalizes well to everyday tasks we develop *Gen2Act* for. It can generate realistic and plausible videos of humans manipulating objects, without introducing significant camera motions/artifacts in the generated videos. We ensure that the image of the scene input to the model doesn't have the robot in the frame (the initial reset position of the robot is such that the arm is mostly out of camera view). The language prompt to the model is of the form "A person `task-name`, static camera" e.g. for the task 'opening the microwave' the input prompt is "A person opening the microwave, static camera."

5.2 Closed-Loop Policy

For each frame in the generated human video $\mathbf{V}_g$ and the robot video $\mathbf{I}_{t-k:k}$, we first extract features, i_g and i_r through a ViT encoder χ. The number of video tokens extracted this way is very large and they are temporally uncorrelated, so we have Transformer encoders Φ_g and Φ_r that process the respective video tokens through gated Cross-Attention Layers based on a Perceiver-Resampler architecture [60] and output a fixed number $N = 64$ of tokens. We use 2 Perceiver-Resampler layers for both the generated video token processing and the robot observation history video processing. These tokens respectively are $z_g = \Phi_g(i_g)$ and $z_r = \Phi_r(i_r)$. During training we sample a fixed sequence of 16 frames from the generated video ensuring that we always sample the first and last frames. For the robot history, we choose the last 8 frames of robot observations. We resize all images to 224x224 dimensions.

We run an off-the-shelf tracking model [61, 21] on the generated video $\mathbf{V}_g$ to obtain tracks τ_g of a random set of points in the first frame P^0. In order to ensure that the latent embeddings from the generated video z_g can distill motion information in the video, we set up a track prediction task conditioned on the video tokens. For this, we define a track prediction transformer $\psi_g(P^0, i_g^0, z_g)$ to predict tracks $\hat{\tau_g}$ and define an auxiliary loss $||\tau_g - \hat{\tau_g}||_2$ to update tokens g_e. Similarly, for the current robot video $\mathbf{I}_{t-k:k}$, we set up a similar track prediction auxiliary loss. We run the ground-truth track prediction once over the entire robot observation sequence (again with random points in the first frame P_0), but during training, the policy is input a chunk of length k in one pass. So here, the track prediction transformer $\psi_r(P^{t-k}, i_{t-k}, r_e^{t-k:t})$ is conditioned on the points in the beginning of the chunk P_{t-k}, the image features at that time-step i^{t-k} and the observation tokens for the chunk z_r. The track prediction transformer has 6 self-attention layers with 8 heads and its role is solely to make the input tokens from generated video / robot observations informative of motion cues. Note that any ground-truth track prediction model can be used for this, and recent advances in point tracking can help improve this step [63, 64]

For ease of prediction, we discretize the action space such that each dimension has 256 bins. So each action dimension can take values in the range $[0, 255]$. The bins are uniformaly distributed within the bounds of each dimension. We predict actions in the end-effector space, and also predict whether to terminate the episode, and whether the gripepr should be open/close. We optimize a Behavior Cloning (BC) objective by minimizing error between the predicted actions $\hat{a}_{t:t+h}$ and the ground-truth $a_{t:t+h}$ through a cross-entropy loss. This discrete action-space for prediction is based on prior works in multi-task imitation learning [1].

Table 3: Comparison of success rates for long-horizon activities via chaining of different tasks. We first obtain sub-tasks for activities with an off-the-shelf LLM and then rollout *Gen2Act* in sequence for the different intermediate tasks.

Activity	**Stages (from Gemini)**	**Success %** Stage 1, Stage 2, Stage 3
Stowing Apple	1. Open the Drawer 2. Place Apple in Drawer 3. Close the Drawer	80, 60, 60
Making Coffee	1. Open the Lid 2. Place K-Cup Pod inside 3. Close the Lid	40, 20, 20
Cleaning Table	1. Pick Tissues from Box 2. Press the Sanitizer Dispenser 3. Wipe the Table with Tissues	60, 40, 40
Heating Soup	1. Open the Microwave 2. Put Bowl inside Microwave 3. Close the Microwave	40, 20, 20

5.3 Chaining Gen2Act for Long-Horizon Manipulation

For deploying *Gen2Act* to solve a manipulation task, we first generate a human video conditioned on the language description of the task and the image of the scene. We then roll out the generated video conditioned closed-loop policy. For chaining *Gen2Act* to perform long-horizon activities consisting of several tasks, we first use an off-the-shelf LLM (e.g. Gemini) to obtain language descriptions of the different tasks. We chain *Gen2Act* for the task sequence by using the last image of the previous policy rollout as the first frame for generating a human video of the subsequent task. We do this chaining in sequence as opposed to generating all the videos from the first image because the final state of the objects in the scene might be different after the robot execution of an intermediate task. This iterative sequencing of video generation after the previous robot execution is made possible by the fast generation capabilities of the Video model we use, VideoPoet (it takes less than 10 seconds to generate a new video after generating the very first video).

We provide examples of the prompt to Gemini and the subsequent outputs for task decomposition below:
Cleaning the Table. Q. What are the steps involved in cleaning a table with sanitizer dispenser and tissues? (write three steps)

A: Steps for Cleaning a Table with Sanitizer Dispenser and Tissues:

- Gather materials: Pick tissues from the box.
- Apply sanitizer: Press the sanitizer dispenser onto the tissue.
- Wipe: Gently wipe the table surface with the sanitizer-dampened tissues.

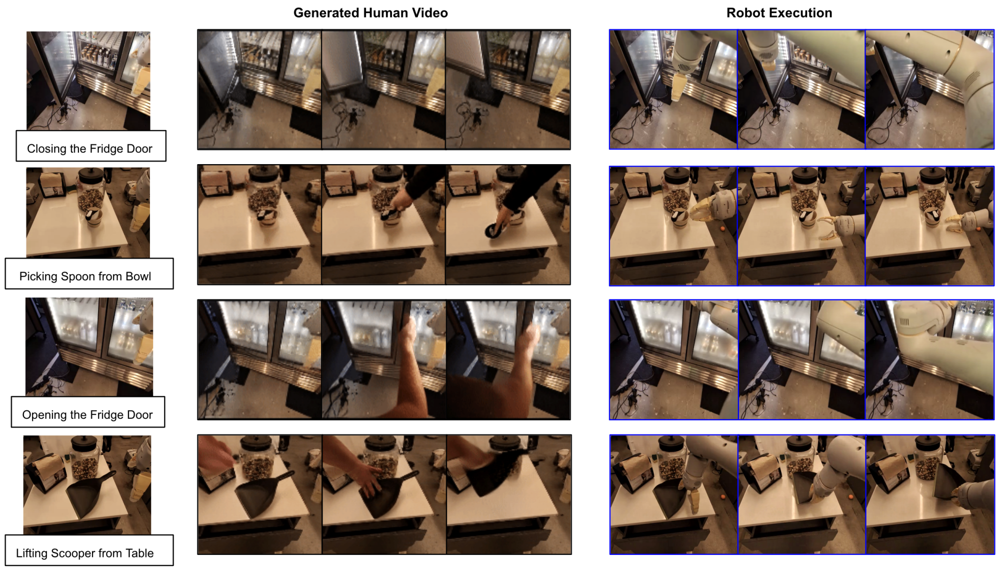

Figure 6: Analysis of failures of *Gen2Act*. The tasks here correspond to object type generalization. We can see that most of the failures of robot execution (top 3 rows) are correlated with incorrect video generations. In the last row the video generation is plausible but the execution is incorrect in following the trajectory of the generated video afetr grasping the object.

Making Coffee. Q. What are the steps involved in making coffee with a Keurig? (write first three steps)

A: Steps for Making Coffee with a Keurig:

- Open the lid: Open the lid of the Keurig machine.
- Insert K-Cup: Place a K-Cup pod into the machine.
- Close the lid: Close the lid of the Keurig machine.

Videos for these tasks are best viewed in the project website.

5.4 Analysis of Failures

Here we discuss the type of failures exhibited by *Gen2Act*. We observe that for MG and to some extent in G, inaccuracies in video generation are less correlated with failures of the policy. While, for the higher levels of generalization, object type (OTG) and motion type (MTG), if video generation yields implausible videos, then the policy doesn't succeed in performing the tasks. This is also evidence that the policy of *Gen2Act* is using the generated human video for inferring motion cues while completing a task, and as such when video generation is incorrect in scenarios where robot data support is limited (e.g. in OTG and MTG), the policy fails. Fig. 6 shows some examples of failures of *Gen2Act* in different tasks. Most of the failures are correlated with video generation (first three rows) but generating a video plausibly (fourth row) is not a guarantee of the policy succeeding because there might be issues with grasping the object correctly and following the trajectory of the object post grasp. This indicates potential for future work to explore recovering more dense motion information from the generated videos beyond point tracks, like object meshes for mitigating some of the failures.