

484 Appendix

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

486 5.1 Human Video Generation

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

497 5.2 Closed-Loop Policy

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

508 We run an off-the-shelf tracking model [61, 21] on the generated video \mathbf{V}_g to obtain tracks τ_g of
509 a random set of points in the first frame P^0 . In order to ensure that the latent embeddings from
510 the generated video z_g can distill motion information in the video, we set up a track prediction task
511 conditioned on the video tokens. For this, we define a track prediction transformer $\psi_g(P^0, i_g^0, z_g)$
512 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
513 current robot video $\mathbf{I}_{t-k:k}$, we set up a similar track prediction auxiliary loss. We run the ground-
514 truth track prediction once over the entire robot observation sequence (again with random points in
515 the first frame P_0), but during training, the policy is input a chunk of length k in one pass. So here,
516 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
517 of the chunk P_{t-k} , the image features at that time-step i^{t-k} and the observation tokens for the chunk
518 z_r . The track prediction transformer has 6 self-attention layers with 8 heads and its role is solely
519 to make the input tokens from generated video / robot observations informative of motion cues.
520 Note that any ground-truth track prediction model can be used for this, and recent advances in point
521 tracking can help improve this step [63, 64]

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

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.

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 supplementary zip.

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.

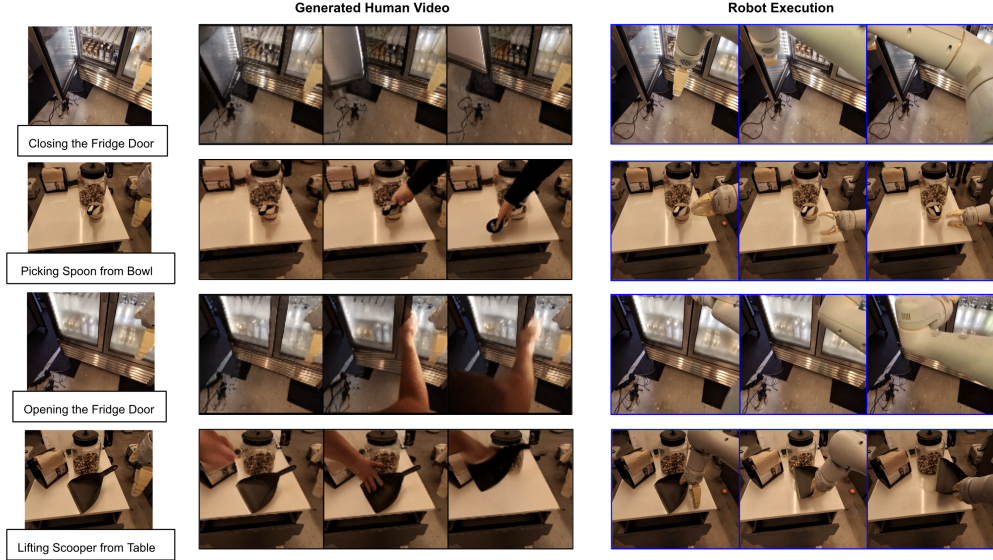


Figure 6: Analysis of failures of *Gen2Act*. The tasks here correspond to object type generalization. We can see that most of the failures of robot execution (top 3 rows) are correlated with incorrect video generations. In the last row the video generation is plausible but the execution is incorrect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

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	<ol style="list-style-type: none"> 1. Open the Drawer 2. Place Apple in Drawer 3. Close the Drawer 	80, 60, 60
Making Coffee	<ol style="list-style-type: none"> 1. Open the Lid 2. Place K-Cup Pod inside 3. Close the Lid 	40, 20, 20
Cleaning Table	<ol style="list-style-type: none"> 1. Pick Tissues from Box 2. Press the Sanitizer Dispenser 3. Wipe the Table with Tissues 	60, 40, 40
Heating Soup	<ol style="list-style-type: none"> 1. Open the Microwave 2. Put Bowl inside Microwave 3. Close the Microwave 	40, 20, 20