Scaling Robot Learning with Semantically Imagined Experience

Tianhe Yu[1], **Ted Xiao**[1], **Austin Stone**[1], **Jonathan Tompson**[1], **Anthony Brohan**[1], **Su Wang**[2], **Jaspiar Singh**[1], **Clayton Tan**[1], **Dee M**[1], **Jodilyn Peralta**[1], **Brian Ichter**[1], **Karol Hausman**[1], **Fei Xia**[1]
[1]Robotics at Google, [2]Google Research
Project website: https://diffusion-rosie.github.io

Abstract: Recent advances in robot learning have shown promise in enabling robots to perform a variety of manipulation tasks and generalize to novel scenarios. One of the key contributing factors to this progress is the scale of robot data used to train the models. To obtain large-scale datasets, prior approaches have relied on either demonstrations requiring high human involvement or engineering-heavy autonomous data collection schemes, both of which are challenging to scale. To mitigate this issue, we propose an alternative route and leverage text-to-image foundation models widely used in computer vision and natural language processing to obtain meaningful data for robot learning without requiring additional robot data. We term our method **Ro**bot Learning with **S**emantically **I**magened **E**xperience **(ROSIE)**. Specifically, we make use of the state of the art text-to-image diffusion models and perform aggressive data augmentation on top of our existing robotic manipulation datasets via inpainting various unseen objects for manipulation, backgrounds, and distractors with text guidance. Through extensive real-world experiments, we show that manipulation policies trained on data augmented this way are able to solve completely unseen tasks with new objects and can behave more robustly w.r.t. novel distractors. In addition, we find that we can improve the robustness and generalization of high-level robot learning tasks such as success detection through training with the diffusion-based data augmentation.

1 Introduction

Though recent progress in robotic learning has shown the ability to learn a number of language-conditioned tasks [1, 2, 3, 4], the generalization properties of such policies is still far less than that of recent large-scale vision-language models [5, 6, 7]. One of the fundamental reasons for these limitations is the lack of diverse data that covers not only a large variety of motor skills, but also a variety of objects and visual domains. This becomes apparent by observing more recent trends in robot learning research – when scaled to larger, more diverse datasets, current robotic learning algorithms have demonstrated promising signs towards more robust and performant robotic systems [1, 2]. However, this promise comes with an arduous challenge: it is difficult to significantly scale up diverse, real-world data collected by robots as it requires either engineering-heavy autonomous schemes such as scripted policies [8, 9] or laborious human teleoperations [10, 2]. To put it into perspective, it took 17 months and 13 robots to collect 130k demonstrations in [2]. In [8], the authors used 7 robots and 16 months to collect 800k autonomous episodes. While some works [11, 12, 13] have proposed potential solutions to this conundrum by generating simulated data to satisfy these robot data needs, they come with their own set of challenges such as generating diverse and accurate enough simulations [1] or solving sim-to-real transfer [14, 15]. Can we find other ways to synthetically generate realistic diverse data without requiring realistic simulations or data collection on real robots?

To investigate this question we look to the field of computer vision. Traditionally, synthetic generation of additional data, whether to improve the accuracy or robustify a machine learning model, has been addressed through data augmentation techniques. These commonly include randomly perturbing the images including cropping, flipping, adding noise, augmenting colors or changing brightness. While

Correspond to {tianheyu,xiafei}@google.com.

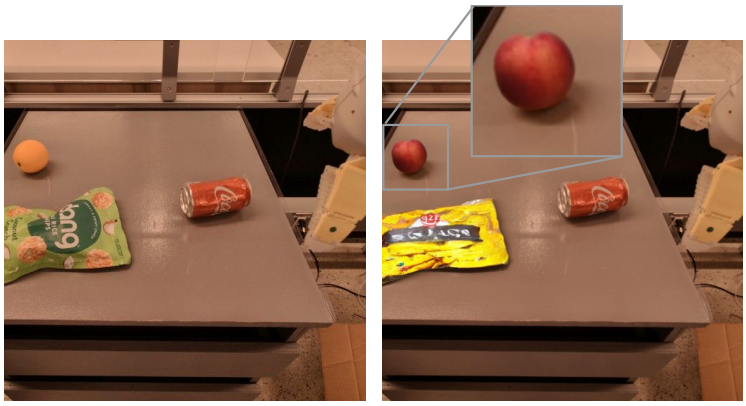


Figure 1: We propose using text-guided diffusion models for data augmentation within the sphere of robot learning. These augmentations can produce highly convincing images suitable for learning downstream tasks. As demonstrated in the figure, some of the objects were produced using our system, and it is difficult to identify which are real and which are generated due to the photorealism of our system.

effective in some computer vision applications, these data augmentation strategies do not suffice to provide novel robotic experiences that can result in a robot mastering a new skill or generalizing to semantically new environments [15, 16, 17]. However, recent progress in high-quality text-to-image diffusion models such as DALL-E 2 [5], Imagen [6] or StableDiffusion [18] provides a new level of data augmentation capability. Such diffusion-based image-generation methods allow us to move beyond traditional data augmentation techniques, for three reasons. First, they can meaningfully augment the semantic aspects of the robotic task through a natural language interface. Second, these methods are built on internet-scale data and thus can be used zero-shot to generate photorealistic images of many objects and backgrounds. Third, they have the capability to meaningfully change only part of the image using methods such as inpainting [19]. These capabilities allow us to generate realistic scenes by incorporating novel distractors, backgrounds, and environments while reflecting the semantics of the new task or scene – essentially distilling the vast knowledge of large generative vision models into robot experience.

As an example, given data for a task such as "move the green chip bag near the orange", we may want to teach the robot to move the chip bag of any colors near many new objects that it has not interacted with, such as "move the yellow chip bag near the peach" (Fig 1). These techniques allow us to exchange the objects from real data for arbitrary relevant objects. Furthermore, they can leave the semantically relevant part of the scene untouched, e.g. the grasp of the chip bag remains, while the orange becomes a peach. This results in a novel, semantically-labelled data point to teach the model a new task. Such a technique can reasonably generate many more examples such as "move the apple near the orange on a wooden desk", "move the plum near the orange", or even "place the coke can in the sink".

In this paper, we investigate how off-the-shelf image-generation methods can vastly expand robot capabilities, enabling new tasks and robust performance. We propose **Ro**bot Learning with **S**emantically **I**magened **E**xperience **(ROSIE)**, a general and semantically-aware data augmentation strategy. ROSIE works by first parsing human provided novel instructions and identifying areas of the scene to alter. It then leverages inpainting to make the necessary alterations, while leaving the rest of the image untouched. This amounts to a *free lunch* of novel tasks, distractors, semantically meaningful backgrounds, and more, as generated by internet-scale-trained generative models. We demonstrate this approach on a large dataset of robotic data and show how a subsequently trained policy is able to perform novel, unseen tasks, and becomes more robust to distractors and backgrounds. Moreover, we show that ROSIE can also improve the robustness of success detection in robotic learning especially in out-of-distribution (OOD) scenarios.

2 Related Work

Scaling robot learning. Given the recent results on scaling data and models in other fields of AI such as language [20, 21, 22] and vision [23, 24, 7], there are multiple approaches trying to do the same in the field of robot learning. One group of methods focuses on scaling up robotic data via simulation [1, 4, 25, 13, 26, 27, 28, 29] with the hopes that the resulting policies and methods will transfer to the real world. The other direction focuses on collecting large diverse datasets in the real world by either teleoperating robots [30, 10, 2, 31] or autonomously collecting data via reinforcement

learning [8, 32, 9] or scripting behaviors [33]. In this work, we present a complementary view on scaling the robot data by making use of state-of-the-art text-conditioned image generation models to enable new robot capabilities, tasks and more robust performance.

Data augmentation and domain randomization. Domain randomization [14, 34, 35] is a common technique for training machine learning models on synthetically generated data. The advantage of domain randomization is that it makes it possible to train models on a wide variety of data to improve generalization. Domain randomization usually involves changing the physical parameters or rendering parameters (lighting, texture, backgrounds) in simulation models [36, 37, 38, 39]. Others use data augmentation to transformer simulated data to be more realistic [15, 16, 40, 41] or vice-versa [42]. Contrary to these methods, we propose to directly augment data collected in the real world. We operate directly on the real-world data and leverage diffusion models to perform photorealistic image manipulation on this data.

Diffusion models for robot control. Though diffusion models [43, 44, 45, 46, 47, 48, 49, 50, 6, 5] have become common-place in computer vision, their application to robotic domains is relatively nascent. Janner et al. [51] uses diffusion models to generate motion plans in robot behavior synthesis. Some works have used the ability of image diffusion models to generate images and perform common sense geometric reasoning to propose goal images fed to object-conditioned policies [52, 53]. The most similar work to ours is CACTI [54], which proposes to use diffusion model for augmenting data collected from the real world via adding new distractors and requires manually provided masks and semantic labels. The recent concurrent work [55] also explores the usage of depth-guided diffusion models for augmenting new tasks and objects in real-world robotic data with human-specified masks and object meshes. In contrast, our work generates both novel distractors and new tasks and demonstrations via *automatically semantically selecting* regions for inpainting with text guidance and generating novel, realistic augmentations.

3 Preliminaries

Diffusion models and inpainting. Diffusion models are a class of generative models that have shown remarkable success in modeling complex distributions [43]. Diffusion models work through an iterative denoising process, transforming Gaussian noise into samples of the distribution guided by a mean squared error loss. Many such models also have the capability for high-quality *inpainting*, essentially filling in masked areas of an image [56, 57, 58, 19]. In addition, such approaches can be guided by language, thus generating areas consistent with both a language prompt and the image as a whole [59].

Multi-task language-conditioned robot learning. Herein we learn vision and language-conditioned robot policies via imitation learning. We denote a dataset $\mathcal{D} \coloneqq \{\mathbf{e}_j\}_{j=1}^{N}$ of N episodes $\mathbf{e} = \{(\mathbf{o}_i, \mathbf{a}_i, \mathbf{o}_{i+1}, \ell)\}_{i=1}^{T}$ where $\mathbf{o}$ denotes the observation, which correspond to the image in our setting, $\mathbf{a}$ denotes the action, and ℓ denotes the language instruction of the episode, identifying the target task. We then learn a policy $\pi(\cdot|\mathbf{o}_i, \ell)$ to generate an action distribution by minimizing the negative-log liklihood of actions, i.e. *behavioral cloning* [60]. To perform large-scale vision-language robot learning, we train the RT-1 architecture [2], which utilizes FiLM-conditioned EfficientNet [61], a TokenLearner [62], and a Transformer [63] to output actions.

4 Robot Learning with Semantically Imagened Experience (ROSIE)

In this section, we introduce our approach, ROSIE, an automated pipeline for scaling up robot data generation via semantic image augmentation. We assume that we have access to episodes of state and action pairs demonstrating a robot executing a task that is labelled with a natural language instruction. As the first step of the pipeline we augment the natural language instruction with a semantically different circumstance. For example, given a demonstration of placing an object in an empty drawer, we add "there is a coke can in the opened drawer". With this natural language prompt, ROSIE generates the mask of the region of interest that is relevant to the language query. Next, given the augmentation text, ROSIE performs inpainting on the selected mask with Imagen Editor [59] to insert semantically accurate objects that follow the augmented text instruction. Importantly, the entire process is applied throughout the robot trajectory, which is now consistently augmented across all the time steps. We present the overview of this pipeline in Fig. 2. We describe the details of each component of ROSIE in the following sections. In Section 4.1, we show how we obtain the mask of the target region using open vocabulary segmentation. In Section 4.2, we discuss two main approaches to proposing prompts used for Imagen Editor, which can be either specified manually or generated automatically with a

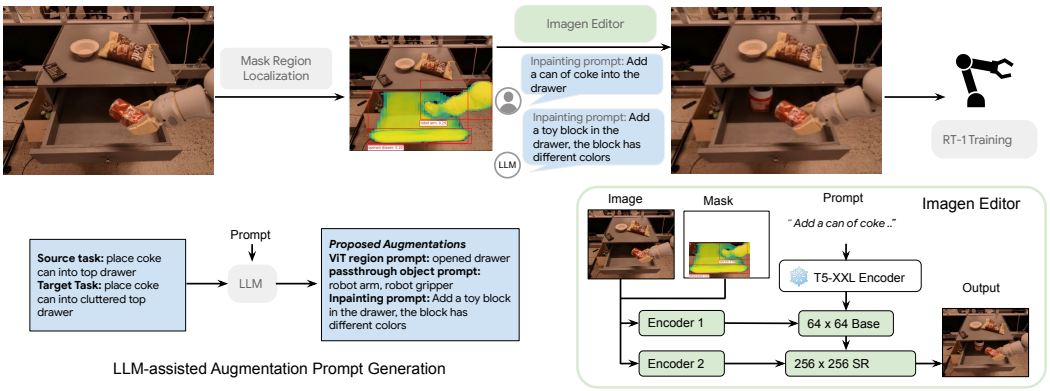


Figure 2: The proposed architecture of ROSIE. First, we localize the augmentation region with open vocabulary segmentation model. Second, we run Imagen Editor to perform text-guided image editing. Finally, we use the augmented data to train an RT-1 manipulation policy [2]. Concretely, we explain ROSIE using the example shown in the figure as follows. We take the original episode with the instruction "place coke can into top drawer" and the goal is to add distractors in the opened drawer to improve the robustness of the policy. For *each image* in the episode, we detect the masks of the open drawer, the robot arm, and the coke can using our first step. We obtain the mask of the target region to add the distractor via subtracting the masks of the robot arm and the coke can that is picked up from the mask of the open drawer. Then, we generate our augmentation proposal leveraging LLMs as described in Section 4.2. We run Imagen Editor with the augmentation text and the selected mask to generate a coke can in the drawer discussed in Section 4.3. We combine both the original episodes and the augmented episodes and perform policy training using multi-task imitation learning.

large language model. In Section 4.3, we discuss how we perform inpainting with Imagen Editor based on the augmentation prompt. Finally, we show how we use the generated data in downstream tasks such as policy learning and learning high-level tasks such as success detection in Section 4.4.

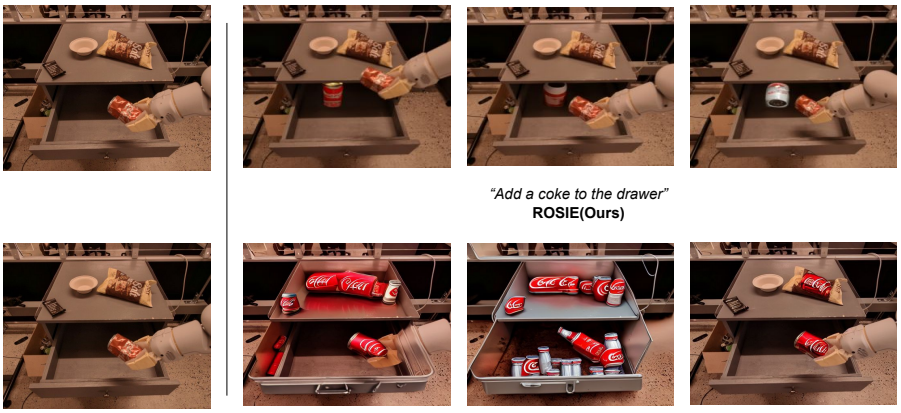


Figure 3: Our augmentation scheme generates more targeted and physically realistic augmentations that are useful for learning downstream tasks, while other text-to-image generation methods such as InstructPix2Pix [64] often makes global changes rendering the image unusable for training.

4.1 Augmentation Region Localization using Open Vocabulary Segmentation

In order to generate semantically meaningful augmentations on top of existing robotic datasets, we first need to detect the region of the image where such augmentation should be performed. To this end, we perform open-vocabulary instance segmentation leveraging the OWL-ViT open-vocabulary detector [65] with an additional instance segmentation head. This additional head predicts fixed resolution instance masks for each bounding box detected by OWL-ViT (similar in style to

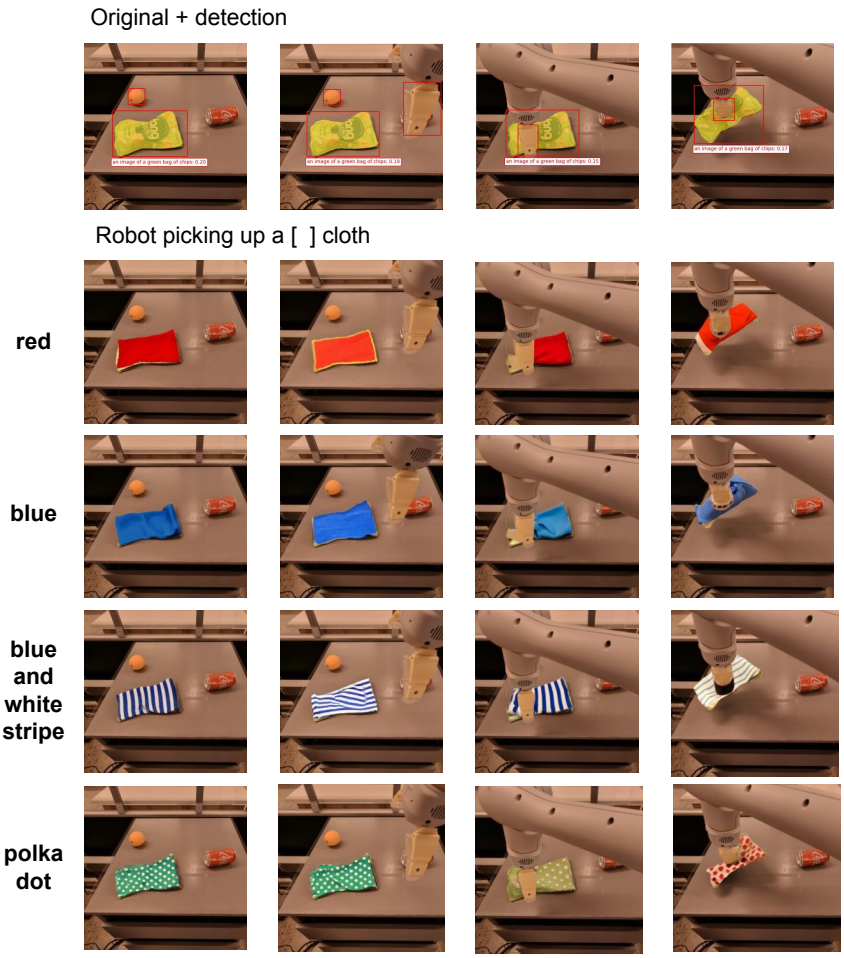


Figure 4: Augmentations of in-hand objects during manipulation. We show examples where ROSIE effectively inpaint novel objects into the original in-hand objects during manipulation. On the top row, we show the original episode with detected masks where the robot picks up the green chip bag. On the following row, we show that ROSIE can inpaint various microfiber cloth with different colors and styles into the original green chip bag. For example, we can simply pass the original episode with the masks and the prompt `Robot picking up a polka dot cloth` to get an episode the robot picking such cloth in a photorealistic manner.

Mask-RCNN [66]). In particular, we freeze the main OWL-ViT model and fine-tune a mask head on Open-Images-V5 instance segmentations [67, 68].

The instance segmentation model provided by OWL-ViT requires a language query that specifies which part of the image should be detected. We can generate masks for objects that the robot arm interacts with. Given each episode $\mathbf{e}$ in our robotic dataset, we first identify the target objects specified in the language instruction ℓ. For example, if ℓ is "pick coke can", the target object of the task is a coke can. We pass the target object as a prompt to the OWL-ViT model to perform segmentation and obtain the resulting mask. We can also generate masks in regions where distractors can be inpainted to improve the robustness of policy. In this setting, we use the OWL-ViT to detect both the table (shown in Figure 2) and all the objects on the table. This allows us to sample a mask on the table in a way that it does not overlap with existing objects (which we call passthrough objects). We provide more examples of masks detected by OWL-ViT from our robotic dataset in Figure 5.

4.2 Augmentation Text Proposal

Next, we discuss two main approaches to attain the augmentation prompt for the text-to-image diffusion model: hand-engineered prompt and LLM-proposed prompt.

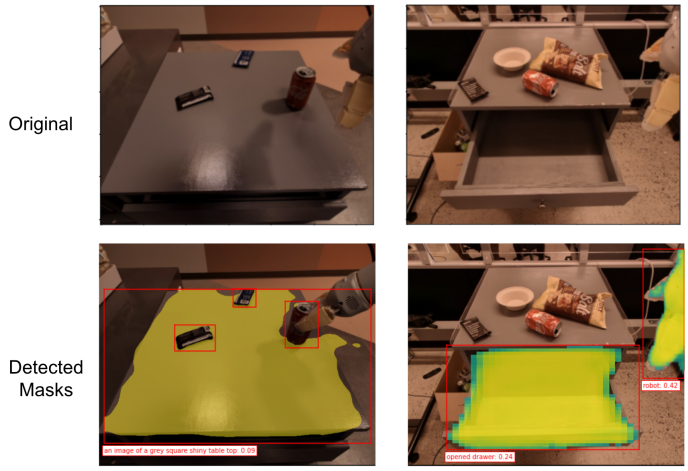


Figure 5: We show the original images from RT-1 datasets on the top row and the images with detected masks and mask labels on the bottom row.

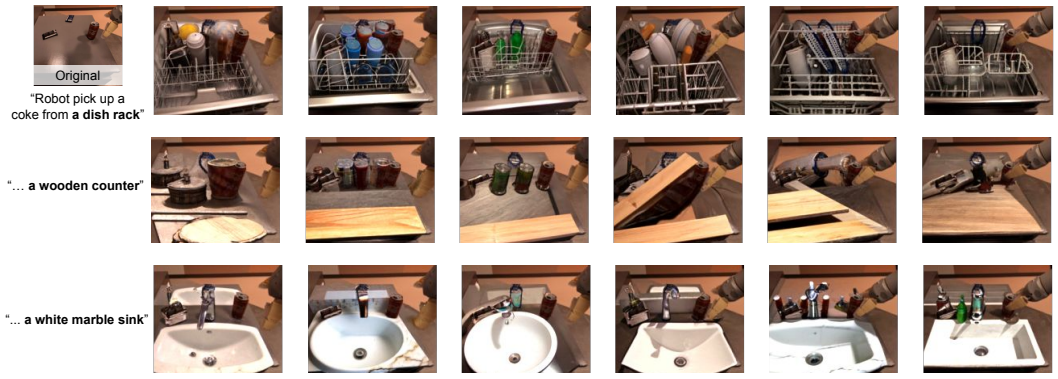
Hand-engineered prompt. The first method involves manually specifying the object to augment. For generating new tasks, we choose objects that lie outside of our training data to ensure that the augmentations are able to expand the data support. For improving robustness of the learned policy and success detection, we randomly pick objects that are semantically meaningful and add them in the prompt to generate meaningful distractors in the scene. For example, in Figure 4 where we aim to generate novel in-hand objects by replacing the original object (green chip bag) with various microfiber cloth, we use the following prompt `Robot picking up a blue and white stripe cloth` to effectively perform inpainting.

LLM-proposed prompt. While hand-engineered prompt may guarantee the generated data to be out-of-distribution, it makes the data generation process less scalable. Therefore, we propose to leverage the power of large language models in proposing objects to augment. We leverage the rich semantics learned in LLMs to propose a vast list of objects with detailed descriptions of visual features for augmentation. We employ GPT-3 [20] as our choice of LLM to propose the augmentation text. In particular, we specify the original task of the episode and the target task after augmentation in the LLM prompt, and ask the LLM to propose the OWL-ViT prompt for detecting masks of both the target region and the passthrough objects. We present an example of LLM-assisted augmentation prompt proposal in Figure 2, where LLM-generated augmentation text is highly informative, which in turn benefits the text-guided image editing. Therefore, we use LLM-proposed prompts in our experiments. Despite that there is some noise in the LLM-proposed prompts (see Appendix C), it generally does not hurt robotic control performance in practice.

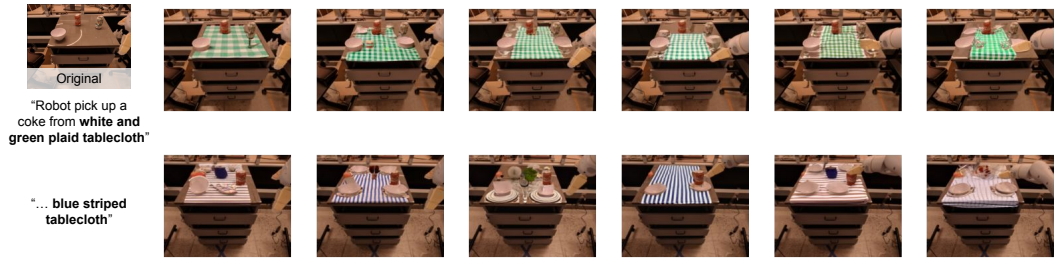
4.3 Diffusion Model for Text-Guided Inpainting

Given the segmentation mask and the augmentation prompt, we perform text-guided image editing via a text-to-image diffusion model. Herein, we use Imagen Editor [59], the latest state-of-the-art text-guided image inpainting model fine-tuned on pre-trained text-to-image generator Imagen [6], though we note that our approach, ROSIE, is agnostic to the choice of inpainting models. Imagen Editor [59] is a cascaded diffusion architecture. All of the diffusion models, i.e., the base model and super-resolution (SR) models (i.e., conditioned on high-resolution 1024×1024 image and mask inputs) are trained with new convolutional image encoders shown in the bottom right corner of Figure 2. Imagen Editor is capable of generating high-resolution photorealistic augmentations, which is crucial for robot learning as it relies on realistic images capturing physical interactions. Moreover, Imagen Editor is trained to de-noise object-oriented masks provided by off-the-shelf object detectors [69] along with random box/stroke masks [70], enabling inpainting with our mask generation procedure.

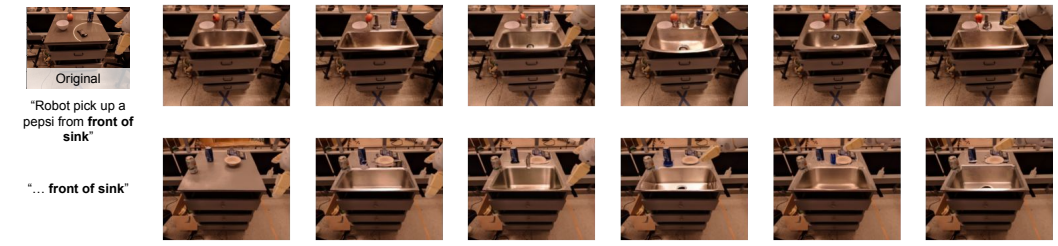
To summarize more formally, given a robotic episode $\mathbf{e}=\{(\mathbf{o}_i,\mathbf{a}_i,\mathbf{o}_{i+1},\ell)\}_{i=1}^{T}$, the mask $\mathbf{m}$ designating the target area(s) to be modified, and our generated augmentation text ℓ_{aug}, we iteratively query Imagen Editor with input $\mathbf{o}_i$, $\mathbf{m}$ and ℓ_{aug} over $i=1,\ldots,T$. As a result, Imagen Editor generates the masked region according to the input text ℓ_{aug} (e.g. inserting novel objects or distractors) while ensuring consistency with the unmasked and unedited content of $\mathbf{o}_i$. This results in generating augmented image $\tilde{\mathbf{o}}_i$. In



(a)



(b)



(c)

Figure 6: We show visualizations of the episodes generated by ROSIE where we replace the regular tabletop in front of the robot with a dish rack, a marble sink and a wooden counter, which never appears in the training dataset. Our results in Section 5.1 and Figure 7 show that the policy trained on such augmentations enables the robot to place objects into a real metal sink.

scenarios where ℓ_{aug} creates a new task, we modify the instruction ℓ to $\tilde{\ell}$. For example, as shown in Figure 4 where we replace the green chip bag with various styles of microfiber cloth, we modify the original instruction $\ell =$ "pick green rice chip bag" to $\tilde{\ell} =$ "pick blue microfiber cloth", pick "polka dot microfiber cloth" and etc. The actions $\{\mathbf{a}_i\}_{i=1}^{T}$ remain unchanged, as Imagen Editor alters novel objects consistently with the semantics of overall image. In summary, ROSIE eventually yields the augmented episode $\tilde{\mathbf{e}} = \{(\tilde{\mathbf{o}}_i, \mathbf{a}_i, \tilde{\mathbf{o}}_{i+1}, \tilde{\ell})\}_{i=1}^{T}$. Powered by the expressiveness of diffusion models and priors learned from internet-scale data, ROSIE is able to provide physically realistic augmentations (e.g. Figure 3) that are valuable in making robot learning more generalizable and robust, which we will show in Section 5.

4.4 Manipulation Model Training

The goal of the augmentation is to improve learning of downstream tasks, e.g. robot manipulation. We train a manipulation policy based on Robotics Transformer (RT-1) architecture [2] discussed in Section 3. Given the ROSIE augmented dataset $\tilde{\mathcal{D}} := \{\tilde{\mathbf{e}}_j\}_{j=1}^{\tilde{N}}$, where $\tilde{N}$ is the number of augmented episodes, we train a policy on top of a pre-trained RT-1 model [2]. The finetuning uses a 1:1 mixing ratio of $\mathcal{D}$ and $\tilde{\mathcal{D}}$. We follow the same training procedure described in [2] except that we use a smaller learning rate 1×10^{-6} to ensure the stability of fine-tuning.

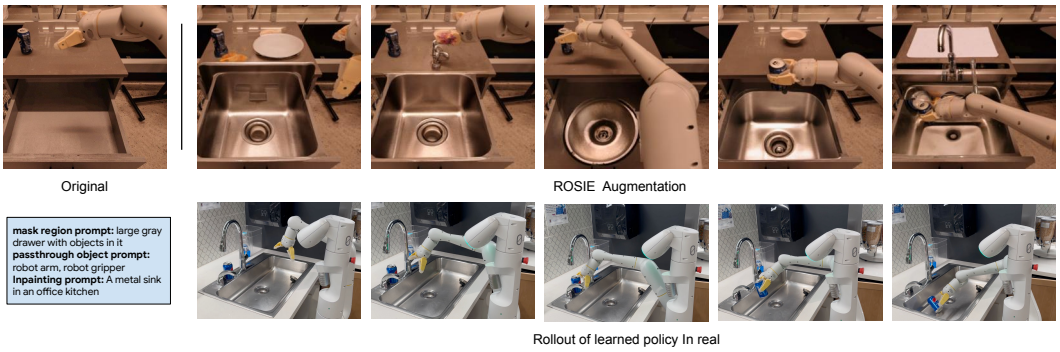


Figure 7: We show an episode augmented by ROSIE (top row) where ROSIE inpaints the metal sink onto the top drawer of the counter and a rollout of policy trained with both the original episodes and the augmented episodes in a real kitchen with a metal sink. The policy successfully performs the task "place pepsi can into sink" even if it is not trained on real data with sink before, suggesting that leveraging the prior of the diffusion models trained with internet-scale data is able to improve generalization of robotic learning in the real world.

5 Experiments

In our experimental evaluation, we focus on robot manipulation and embodied reasoning (e.g. detecting if a manipulation task is performed successfully). We design experiments to answer the following research questions:

1. **RQ1**: Can we leverage semantic-aware augmentation to learn completely new skills only seen through diffusion models?
2. **RQ2**: Can we leverage semantic-aware augmentation to make our policy more robust to visual distractors?
3. **RQ3**: Can we leverage semantic-aware augmentation to bootstrap high-level embodied reasoning such as success detection?

To answer these questions, we perform empirical evaluations of ROSIE using the multi-task robotic dataset collected in [2], which consists of $\sim$130k robot demonstrations with 744 language instructions collected in laboratory offices and kitchens. These tasks include skills such as picking, placing, opening and closing drawers, moving objects near target containers, manipulating objects into or out of the drawers, and rearranging objects. For more details regarding the tasks and the data used we refer to Brohan et al. [2].

In our experiments, we aim to understand the effects of both the augmented text and the augmented images on policy learning. We thus perform two comparisons, ablating these changes:

1. **Pre-trained RT-1 (NoAug)**: we take the RT-1 policy trained on the 744 tasks in [2]. While pre-trained RT-1 is not trained on tasks with the augmentation text and generated objects, it has been shown to enjoy promising pre-training capability and demonstrate excellent zero-shot generalization to unseen scenarios [2] and therefore, should have the ability to tackle the novel tasks to some extent.
2. **Fine-tuned RT-1 with Instruction Augmentation (InstructionAug)**: Similar to Xiao et al. [71], we relabel the original episodes in RT-1 dataset to new instructions generated via our augmentation text proposal 4.2 while keeping the images unchanged. We expect this method to bring the text instructions in-distribution but fail to recognize the visuals of the augmented objects.

For implementation details and hyperparameters, please see Appendix A.

5.1 RQ1: Learning new skills

To answer RQ1, we augment the RT-1 dataset via generating new objects that the robot needs to manipulate. We evaluate our method and the baselines in the following four categories with increasing level of difficulty.

Learning to move objects near generated novel containers First, we test the tasks of moving training objects near unseen containers. We visualize such unseen containers in Figure 10 in Appendix B. We select the tasks "move {some object} near white bowl" and "move {some object} near paper bowl" within the RT-1 dataset, which yields 254 episodes in total. We use the augmentation text proposals to replace the white bowl and the paper bowl with the following list of objects {lunch box, woven basket, ceramic pot, glass mason jar, orange paper plate}, which are visualized in Figure 10. For each augmentation, we augment the same number of episodes as the original task.

As shown in Table 1, our ROSIE fine-tuned RT-1 policy (trained on both the whole RT-1 training set of 130k episodes and the generated novel tasks) outperforms pre-trained RT-1 policy and fine-tuned RT-1 with instruction augmentations, suggesting that ROSIE is able to generate fully unseen tasks that are beneficial for control and exceeds the inherent transfer ability of RT-1.

Learning to place objects into generated unseen containers Second, we perform a similar experiment, where we focus on *placing* objects into the novel target containers, rather than just nearby. Example augmentations are shown in Figure 10. Table 1 again shows ROSIE outperforms both pre-trained RT-1 and RT-1 with instruction augmentation by at least 75%.

Learning to grasp generated unknown deformable objects Third, we test the limits of ROSIE on novel tasks where the object to be manipulated is generated via ROSIE. We pick the set of tasks "pick green chip bag" from the RT-1 dataset consisting of 1309 episodes. To accurately generate the mask of the chip bag throughout the trajectory, we run our open-vocabulary segmentation to detect the chip bag and the robot gripper as the passthrough objects so that we can filter out the robot gripper to obtain the accurate mask of the chip bag when it is grasped. We further query Imagen Editor to substitute the chip bag with a fully unknown microfiber cloth with distinctive colors (black and blue), with augmentations shown in Figure 4. Table 1 again demonstrates that ROSIE outperforms pre-trained RT-1 and RT-1 with instruction augmentation by at least 150%, proving that ROSIE is able to expand the manipulation task family via diversifying the manipulation targets and boost the policy performance in the real world.

Learning to place objects into an unseen kitchen sink in a new background Finally, to further stress-test our diffusion-based augmentation pipeline, we try to learn to place object into a sink. Note that the robot has never collected data for that task in the real world. We generate a challenging scenario where we take the all the RT-1 tasks that perform placing a can into the top drawer of a counter (779 episodes in total) and deploy ROSIE to detect the open drawer and replace the drawer with a metal sink using Imagen Editor (see the first row of Figure 7 for the visualization). Similar to the above two experiments, we dynamically compute the mask of the open drawer at each frame of the episode while removing the robot arm and the can in the robot hand from the mask. Note that the generated sink makes the scene completely out of the training distribution, which poses considerable difficulty to the pre-trained RT-1 policy. The results in the last row in Table 1 confirm this. ROSIE achieves 60% overall success rate in placing the coke can and the pepsi can into the sink whereas the RT-1 policy is not able to locate the can and fails to achieve any success. In Figure 7, we include the visualizations of a trajectory of the original episode with augmentations that replaces the drawer with the sink and a trajectory of the policy rollout performing the task near a real metal sink. Our method effectively learns from the episodes with the sink generated by ROSIE and completes the task that involve the sink in the real kitchen.

Overall, through these experiments, ROSIE is shown to be capable of effectively inpainting both the objects that require rich manipulation and the target object of the manipulation policy, significantly augmenting the number of tasks in robotic manipulation. These results indicate a promising path to scaling robot learning without extra effort of real data collection.

5.2 RQ2: Robustifying manipulation policies

We investigate RQ2 with two scenarios: policy robustness w.r.t. different backgrounds and new distractors.

Unseen background. We employ ROSIE to augment the background in our training data. We perform two types of augmentations: replacing the table top with a colorful table cloth and inserting a sink on the table top. We select two manipulation tasks, "pick coke can" and "pick pepsi can" from our training set, which consists of 1222 episodes in total. We run open-vocabulary segmentation to detect the table and passthrough objects, which consist of the robot arm and the target can. To generate a diverse set of table cloth during augmentation, we query GPT-3 with the following prompt:

Task Family / Text Instruction	NoAug	InstructionAug	ROSIE
Move object near novel object	0.86	0.78	**0.94**
move coke can/orange near lunch box	0.8	0.6	0.9
move coke can/orange near woven basket	0.7	0.6	0.9
move coke can/orange near ceramic pot	1.0	0.9	1.0
move coke can/orange near glass mason jar	0.9	0.8	1.0
move coke can/orange near orange paper plate	0.9	1.0	0.9
Pick up novel object	0.25	0.3	**0.75**
pick blue microfiber cloth	0.1	0.4	0.8
pick black microfiber cloth	0.4	0.2	0.7
Place object into novel container	0.13	0.25	**0.44**
place coke can into orange plastic plate	0.0	0.19	0.5
place coke can into blue plastic plate	0.25	0.06	0.38
Place object into sink	0.0	-	**0.6**
place coke can into sink	0.0	-	0.8
place pepsi can into sink	0.0	-	0.4
Pick up object in new backgrounds	0.33	-	**0.71**
pick coke can on an orange table cloth	0.0	-	0.4
pick pepsi can on an orange table cloth	0.0	-	0.7
pick coke can on an blue and white table cloth	0.2	-	0.7
pick pepsi can on an blue and white table cloth	0.8	-	0.8
pick coke can near the side of a sink	0.4	-	0.5
pick pepsi can near the side of a sink	0.3	-	0.7
pick coke can in front of a sink	0.4	-	0.9
pick pepsi can in front of a sink	0.5	-	1.0
Place object into cluttered drawer	0.38	-	**0.55**
place blue chip bag into top drawer	0.5	-	0.4
place green jalapeno chip bag into top drawer	0.4	-	0.5
place green rice chip bag into top drawer	0.4	-	0.5
place brown chip bag into top drawer	0.2	-	0.8
Pick up object (with OOD distractors)	0.33	-	**0.37**
pick coke can	0.33	-	0.37

Table 1: Full Experimental Results for ROSIE. The blue shaded results correspond to RQ1 and the orange shaded results correspond to RQ2. For each task family from top to the bottom, we performed evaluations with 50, 20, 16, 10, 80, 40, and 27 episodes respectively (243 episodes in total). ROSIE outperforms **NoAug** (pre-trained RT-1 policy) and **InstructionAug** (fine-tuned RT-1 policy with instruction augmentation [71]) in both categories, suggesting that ROSIE can significantly improve the generalization to novel tasks and robustness w.r.t. different distractors.

```
inpainting prompt:  pick coke can from a red and yellow table cloth
goal:  list 30 more table cloth with different vivid colors and styles with visual details
inpainting prompt:  pick coke can from
1.  Navy blue and white striped table cloth
2.  White and pink polka dot table cloth
3.  Mint green and light blue checkered table cloth
4.  Cream and gray floral table cloth
5.  Hot pink and red floral table cloth
...
```

We show the some example answers from GPT-3 in blue, which are semantically meaningful. We use Imagen Editor to replace the table top except the target can with the LLM-proposed table cloth. To inpaint a sink on the table, we follow the same procedure described in the placing objects into unseen sink task in Section 5.1 except that we inpaint the sink on the table top rather than the open drawer. We present visualizations of such augmentations in Figure 6. We fine-tune the pre-trained RT-1 policy on both the

original data and the augmented episodes with generated table cloth and metal sink. As shown in Table 1, ROSIE + RT-1 signifcantly outperforms RT-1 **NoAug** in 7 out of 8 settings while performing similarly to **NoAug** in the remaining scenario, achieving an overall 115% improvement. Therefore, ROSIE is highly effectively in robustifying policy performance under varying table textures and background.

Novel distractors. To test whether ROSIE can improve policy robustness w.r.t. novel distractors and cluttered scenes, we consider the following two tasks. First, we train a policy solely from the task "pick coke can" and investigate its ability to perform this task with distractor coke cans, which have not been seen in the 615 training episodes. To this end, we employ ROSIE to add an equal number of augmented episodes with additional coke cans on the table (see Figure 8 in Appendix B for visualizations). As shown in Table 1, RT-1 + ROSIE augmentations improves the performance over RT-1 trained with "pick coke can" data only in scenarios where there are multiple coke cans on the table.

Second, we evaluate a task that places a chip bag into a drawer and investigate its ability to perform this task with distractor objects already in the drawer, also unseen during training. This scenario is challenging for RT-1, since the distractor object in the drawer will confuse the model and make it more likely to directly output termination action. We use ROSIE to add novel objects to the drawer, as shown in Figure 9 in Appendix B and follow the same training procedure as in the coke can experiment. Table 1 shows that RT-1 trained with both the original data and ROSIE generated data outperforms RT-1 with only original data. Our interpretation is that RT-1 trained from the training data never sees this situation before and it incorrectly believes that the task is already solved at the first frame, whereas ROSIE can mitigate this issue via expanding the dataset using generative models.

5.3 RQ3: A Case Study on Success Detection

In this section, we show that ROSIE is also effective in improving high-level robotic embodied reasoning tasks, such as success detection. Success detection (or failure detection) is an important capability for autonomous robots for accomplishing tasks in dynamic situations that may require adaptive feedback from the environment. Given large diversity of potential situations that a robot might encounter, a general solution to this problem may involve deploying learned failure detection systems [72] that can improve with more data. As recent work [71] has shown, visual-language models (VLMs) such as CLIP [73] with internet scale pre-training can be fine-tuned on domain specific robotic experience to perform embodied reasoning such as success detection. However, collecting domain specific fine-tuning data is often expensive, and it is difficult to scale data collection to cover all potential success and failure cases. This challenge is similar to the one of learning a robust policy that we presented in the previous sections, where the dataset of robot data might include data distribution biases that are difficult to correct with on-robot data collection alone.

As a motivating example, consider the experimental setting from Section 5.1 where a large dataset of teleoperated demonstrations was collected for placing various household objects into empty cabinet drawers. A success detector trained on this dataset would require additional priors and/or data to generalize to images of cluttered drawers.

To study this setting, we utilize ROSIE to augment 22764 episodes of placing objects into drawers tasks from the dataset used in [71] and then fine-tune a CLIP-based success detector following the procedure in [71]. Starting from the episodes of robotic placing into empty drawers, we create two augmented datasets with ROSIE to emulate visual clutter: one dataset **(A)** that includes generated distractor chip bags inside the drawer and one dataset **(B)** that includes generated soda cans inside the drawer. Both datasets have the same number of episodes as the original dataset. We evaluate the fine-tuned CLIP-based success detector with and without ROSIE-augmented episodes in two datasets: the in-distribution set and the OOD set. Our in-distribution set contains 76 episodes of robot putting green rice chip bag into the drawer and taking it out of the drawer, while the OOD set contains 58 episodes of robot putting (green rice, green japaleno, blue, brown) chip bag into the drawer, but the drawer contains other items, which are not observed in the training set. Note that this OOD set makes success detection particularly challenging as the model can easily be misguided by the cluttered distractors in the drawer and make incorrect predictions even if the robot fails to place the target object into the drawer.

By utilizing increasing amounts of augmentation from ROSIE, we find that learned success detectors become increasingly robust detecting successes and failures in real-world difficult cluttered OOD drawer scenarios in terms of F1 score, as seen in Table 2. Note that our OOD dataset is highly challenging, as discussed above, so that the prior work [71] without augmentations struggles a lot in this setting whereas ROSIE obtains a reasonable performance. Furthermore, we find that the accuracy

	No Aug	ROSIE Aug (**A**)	ROSIE Aug ((**A**) + (**B**))
Overall	0.43	0.56	**0.62**
In-Distribution set	0.66	**0.67**	0.66
OOD set	0.19	0.45	**0.57**

Table 2: CLIP success detection Results. ROSIE improves the robustness of the success detection on hard OOD cases as the number of augmentations increases. All numbers are the F1 score and we use 0.5 as the threshold. We augment the data with datasets **A** and **B**, which include different distractors as described in text.

on the standard, in-distribution tasks remains unchanged. This indicates that ROSIE can be used as a general semantically-consistent data augmentation technique across various tasks such as policy learning and embodied reasoning.

6 Societal Impact

The model used in this work is a text-guided image generation model, which open many new possibilities for content creation and subsequently many risks. Our approach attempts to minimize many of these risks through a controlled usage of these technologies, by only modifying local patches of images and using narrowly scoped semantic labels. We further follow accepted responsible AI practices, such as regularly inspecting data before training on it, and in general recommend researchers to establish robust inspection and filtering mechanisms when utilizing text-guided image generation models for data augmentation.

7 Discussion, Future Work, and Conclusion

In summary, we presented ROSIE, a system that uses off-the-shelf text-guided image generation models to vastly expand robotics datasets without any real-world data collection. To accomplish this, we generated new instructions and their corresponding text prompts for alternating the images, enabling robots to achieve tasks that were *only seen through the lens of image generation process*. We were also able to generate semantically meaningful augmentations of the images, enabling various learned models trained on the data to be more robust with respect to OOD scenes. Lastly, we experimentally validated the proposed method on a variety of language-conditioned manipulation tasks.

Though the method is general and flexible, there are a few limitations of this work that we aim to address in the future. First, we only augment the appearance of the objects and scenes, and do not generate new motions. To alleviate this limitation of not augmenting physics and motions, we could consider mixing in simulation data as a potential source of diverse motion data. Another limitation of the proposed method is that it performs image augmentation per frame, which can lead to a loss in temporal consistency. However, we find that at least for the architecture that we use (Robotics Transformer [2]), we do not suffer from a performance drop. State of the art text-to-video diffusion models [74, 75, 76, 77] can generate temporally consistent videos but might lose photorealism and physics realism. We speculate that this can cause downstream task learning performance to deteriorate. The trade off between photorealism and temporally consistency remains an interesting topic for future studies. Finally, we use a diffusion model for image augmentation, which is computationally heavy and limits our capability to perform on-the-fly augmentation. As a future direction, we could consider other models such as the mask transformer-based architecture [78], which is 10x more efficient.

Acknowledgments

We would like to acknowledge Sharath Maddineni, Brianna Zitkovich, Vincent Vanhoucke, Kanishka Rao, Quan Vuong, Alex Irpan, Sarah Laszlo, Bob Wei, Sean Kirmani, Pierre Sermanet and the greater teams at Robotics at Google for their feedback and contributions.

References

[1] Y. Jiang, A. Gupta, Z. Zhang, G. Wang, Y. Dou, Y. Chen, L. Fei-Fei, A. Anandkumar, Y. Zhu, and L. Fan. Vima: General robot manipulation with multimodal prompts. *arXiv preprint arXiv:2210.03094*, 2022.

[2] 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.

[3] M. Shridhar, L. Manuelli, and D. Fox. Cliport: What and where pathways for robotic manipulation. In *Conference on Robot Learning*, 2022.

[4] M. Shridhar, L. Manuelli, and D. Fox. Perceiver-actor: A multi-task transformer for robotic manipulation. *arXiv preprint arXiv:2209.05451*, 2022.

[5] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen. Hierarchical Text-Conditional Image Generation with CLIP Latents. In *arXiv:2204.06125*, 2022.

[6] 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.

[7] X. Chen, X. Wang, S. Changpinyo, A. Piergiovanni, P. Padlewski, D. Salz, S. Goodman, A. Grycner, B. Mustafa, L. Beyer, et al. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*, 2022.

[8] D. Kalashnikov, J. Varley, Y. Chebotar, B. Swanson, R. Jonschkowski, C. Finn, S. Levine, and K. Hausman. Mt-opt: Continuous multi-task robotic reinforcement learning at scale. *arXiv preprint arXiv:2104.08212*, 2021.

[9] A. X. Lee, C. M. Devin, Y. Zhou, T. Lampe, K. Bousmalis, J. T. Springenberg, A. Byravan, A. Abdolmaleki, N. Gileadi, D. Khosid, et al. Beyond pick-and-place: Tackling robotic stacking of diverse shapes. In *5th Annual Conference on Robot Learning*, 2021.

[10] 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.

[11] F. Xia, A. R. Zamir, Z. He, A. Sax, J. Malik, and S. Savarese. Gibson env: Real-world perception for embodied agents. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9068–9079, 2018.

[12] E. Kolve, R. Mottaghi, W. Han, E. VanderBilt, L. Weihs, A. Herrasti, D. Gordon, Y. Zhu, A. Gupta, and A. Farhadi. Ai2-thor: An interactive 3d environment for visual ai. *arXiv preprint arXiv:1712.05474*, 2017.

[13] M. Savva, A. Kadian, O. Maksymets, Y. Zhao, E. Wijmans, B. Jain, J. Straub, J. Liu, V. Koltun, J. Malik, et al. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9339–9347, 2019.

[14] B. Mehta, M. Diaz, F. Golemo, C. J. Pal, and L. Paull. Active domain randomization. In *Conference on Robot Learning*, pages 1162–1176. PMLR, 2020.

[15] F. Sadeghi, A. Toshev, E. Jang, and S. Levine. Sim2real view invariant visual servoing by recurrent control. *arXiv preprint arXiv:1712.07642*, 2017.

[16] I. Akkaya, M. Andrychowicz, M. Chociej, M. Litwin, B. McGrew, A. Petron, A. Paino, M. Plappert, G. Powell, R. Ribas, et al. Solving rubik's cube with a robot hand. *arXiv preprint arXiv:1910.07113*, 2019.

[17] M. Laskin, A. Srinivas, and P. Abbeel. Curl: Contrastive unsupervised representations for reinforcement learning. In *International Conference on Machine Learning*, pages 5639–5650. PMLR, 2020.

[18] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-Resolution Image Synthesis with Latent Diffusion Models. In *CVPR*, 2022.

[19] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5505–5514, 2018.

[20] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[21] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[22] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

[23] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[24] 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. *arXiv preprint arXiv:2204.14198*, 2022.

[25] M. Shridhar, J. Thomason, D. Gordon, Y. Bisk, W. Han, R. Mottaghi, L. Zettlemoyer, and D. Fox. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10740–10749, 2020.

[26] S. James, Z. Ma, D. R. Arrojo, and A. J. Davison. Rlbench: The robot learning benchmark & learning environment. *IEEE Robotics and Automation Letters*, 5(2):3019–3026, 2020.

[27] T. Yu, D. Quillen, Z. He, R. Julian, K. Hausman, C. Finn, and S. Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on robot learning*, pages 1094–1100. PMLR, 2020.

[28] M. Mittal, C. Yu, Q. Yu, J. Liu, N. Rudin, D. Hoeller, J. L. Yuan, P. P. Tehrani, R. Singh, Y. Guo, et al. Orbit: A unified simulation framework for interactive robot learning environments. *arXiv preprint arXiv:2301.04195*, 2023.

[29] F. Xiang, Y. Qin, K. Mo, Y. Xia, H. Zhu, F. Liu, M. Liu, H. Jiang, Y. Yuan, H. Wang, et al. Sapien: A simulated part-based interactive environment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11097–11107, 2020.

[30] A. Mandlekar, J. Booher, M. Spero, A. Tung, A. Gupta, Y. Zhu, A. Garg, S. Savarese, and L. Fei-Fei. Scaling robot supervision to hundreds of hours with roboturk: Robotic manipulation dataset through human reasoning and dexterity. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1048–1055. IEEE, 2019.

[31] F. Ebert, Y. Yang, K. Schmeckpeper, B. Bucher, G. Georgakis, K. Daniilidis, C. Finn, and S. Levine. Bridge data: Boosting generalization of robotic skills with cross-domain datasets. *arXiv preprint arXiv:2109.13396*, 2021.

[32] 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.

[33] S. Dasari, F. Ebert, S. Tian, S. Nair, B. Bucher, K. Schmeckpeper, S. Singh, S. Levine, and C. Finn. Robonet: Large-scale multi-robot learning. *arXiv preprint arXiv:1910.11215*, 2019.

[34] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 23–30. IEEE, 2017.

[35] J. Tremblay, A. Prakash, D. Acuna, M. Brophy, V. Jampani, C. Anil, T. To, E. Cameracci, S. Boochoon, and S. Birchfield. Training deep networks with synthetic data: Bridging the reality gap by domain randomization. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 969–977, 2018.

[36] M. Laskin, K. Lee, A. Stooke, L. Pinto, P. Abbeel, and A. Srinivas. Reinforcement learning with augmented data. *Advances in neural information processing systems*, 33:19884–19895, 2020.

[37] I. Kostrikov, D. Yarats, and R. Fergus. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. *arXiv preprint arXiv:2004.13649*, 2020.

[38] N. Hansen, R. Jangir, Y. Sun, G. Alenyà, P. Abbeel, A. A. Efros, L. Pinto, and X. Wang. Self-supervised policy adaptation during deployment. *arXiv preprint arXiv:2007.04309*, 2020.

[39] B. Li, V. François-Lavet, T. Doan, and J. Pineau. Domain adversarial reinforcement learning. *arXiv preprint arXiv:2102.07097*, 2021.

[40] K. Rao, C. Harris, A. Irpan, S. Levine, J. Ibarz, and M. Khansari. Rl-cyclegan: Reinforcement learning aware simulation-to-real. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11157–11166, 2020.

[41] D. Ho, K. Rao, Z. Xu, E. Jang, M. Khansari, and Y. Bai. Retinagan: An object-aware approach to sim-to-real transfer. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10920–10926. IEEE, 2021.

[42] S. James, P. Wohlhart, M. Kalakrishnan, D. Kalashnikov, A. Irpan, J. Ibarz, S. Levine, R. Hadsell, and K. Bousmalis. Sim-to-real via sim-to-sim: Data-efficient robotic grasping via randomized-to-canonical adaptation networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12627–12637, 2019.

[43] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

[44] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[45] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[46] Y. Song and S. Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020.

[47] J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[48] A. Q. Nichol and P. Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021.

[49] P. Dhariwal and A. Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

[50] J. Ho and T. Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[51] M. Janner, Y. Du, J. B. Tenenbaum, and S. Levine. Planning with diffusion for flexible behavior synthesis. *arXiv preprint arXiv:2205.09991*, 2022.

[52] W. Liu, T. Hermans, S. Chernova, and C. Paxton. Structdiffusion: Object-centric diffusion for semantic rearrangement of novel objects. *arXiv preprint arXiv:2211.04604*, 2022.

[53] I. Kapelyukh, V. Vosylius, and E. Johns. Dall-e-bot: Introducing web-scale diffusion models to robotics. *arXiv preprint arXiv:2210.02438*, 2022.

[54] 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.

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

[56] A. A. Efros and W. T. Freeman. Image quilting for texture synthesis and transfer. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 341–346, 2001.

[57] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016.

[58] S. Iizuka, E. Simo-Serra, and H. Ishikawa. Globally and locally consistent image completion. *ACM Transactions on Graphics (ToG)*, 36(4):1–14, 2017.

[59] S. Wang, C. Saharia, C. Montgomery, J. Pont-Tuset, S. Noy, S. Pellegrini, Y. Onoe, S. Laszlo, D. J. Fleet, R. Soricut, et al. Imagen editor and editbench: Advancing and evaluating text-guided image inpainting. *arXiv preprint arXiv:2212.06909*, 2022.

[60] D. A. Pomerleau. Alvinn: An autonomous land vehicle in a neural network. *Advances in neural information processing systems*, 1, 1988.

[61] M. Tan and Q. Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6105–6114. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/tan19a.html.

[62] M. Ryoo, A. Piergiovanni, A. Arnab, M. Dehghani, and A. Angelova. Tokenlearner: Adaptive space-time tokenization for videos. *Advances in Neural Information Processing Systems*, 34: 12786–12797, 2021.

[63] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[64] T. Brooks, A. Holynski, and A. A. Efros. Instructpix2pix: Learning to follow image editing instructions. *arXiv preprint arXiv:2211.09800*, 2022.

[65] M. Minderer, A. Gritsenko, A. Stone, M. Neumann, D. Weissenborn, A. Dosovitskiy, A. Mahendran, A. Arnab, M. Dehghani, Z. Shen, et al. Simple open-vocabulary object detection with vision transformers. *arXiv preprint arXiv:2205.06230*, 2022.

[66] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017. doi:10.1109/ICCV.2017.322.

[67] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, T. Duerig, and V. Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv:1811.00982*, 2018.

[68] R. Benenson, S. Popov, and V. Ferrari. Large-scale interactive object segmentation with human annotators. In *CVPR*, 2019.

[69] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L. Chen. Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. *CoRR*, abs/1801.04381, 2018. URL http://arxiv.org/abs/1801.04381.

[70] R. Suvorov, E. Logacheva, A. Mashikhin, A. Remizova, A. Ashukha, A. Silvestrov, N. Kong, H. Goka, K. Park, and V. Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. *arXiv preprint arXiv:2109.07161*, 2021.

[71] T. Xiao, H. Chan, P. Sermanet, A. Wahid, A. Brohan, K. Hausman, S. Levine, and J. Tompson. Robotic skill acquisition via instruction augmentation with vision-language models. *arXiv preprint arXiv:2211.11736*, 2022.

[72] D. Kalashnkov, J. Varley, Y. Chebotar, B. Swanson, R. Jonschkowski, C. Finn, S. Levine, and K. Hausman. Mt-opt: Continuous multi-task robotic reinforcement learning at scale. *arXiv*, 2021.

[73] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[74] J. Ho, W. Chan, C. Saharia, J. Whang, R. Gao, A. Gritsenko, D. P. Kingma, B. Poole, M. Norouzi, D. J. Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022.

[75] U. Singer, A. Polyak, T. Hayes, X. Yin, J. An, S. Zhang, Q. Hu, H. Yang, O. Ashual, O. Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022.

[76] R. Villegas, M. Babaeizadeh, P.-J. Kindermans, H. Moraldo, H. Zhang, M. T. Saffar, S. Castro, J. Kunze, and D. Erhan. Phenaki: Variable length video generation from open domain textual description. *arXiv preprint arXiv:2210.02399*, 2022.

[77] E. Molad, E. Horwitz, D. Valevski, A. R. Acha, Y. Matias, Y. Pritch, Y. Leviathan, and Y. Hoshen. Dreamix: Video diffusion models are general video editors, 2023. URL https://arxiv.org/abs/2302.01329.

[78] H. Chang, H. Zhang, J. Barber, A. Maschinot, J. Lezama, L. Jiang, M.-H. Yang, K. Murphy, W. T. Freeman, M. Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023.

Appendices

A Experiment Details

A.1 Implementation Details and Hyperparameters

We take a pre-trained RT-1 policy with 35M parameters and trained for 315k steps at a learning rate of 1×10^{-4} and fine-tune the RT-1 policy with 1:1 mixing ratio of the original 130k episodes of RT-1 data and the ROSIE-generated episodes with for 85k steps with learning rate 1×10^{-6}. We follow all the other policy training hyperparameters used in [2].

To obtain the accurate segmentation mask of the target region of augmentations, we set a threshold for filtering out predicted masks with low prediction scores of both the region of the interest and passthrough objects given by OWL-ViT. In cases where we have multiple detected masks, we always select the one with highest prediction score. Specifically, for experiments where the robot is required to pick novel objects or place objects into novel containers or move objects near unseen containers (Section 5.1), we use a threshold of 0.07 to detect the in-hand objects and the containers while using a threshold of 0.05 to detect passthrough objects, which are the robot arm and robot gripper. In experiments where the robot is instructed to place the coke can or the pepsi can into the unknown sink or pick up coke can and the pepsi can with new background , we use a threshold of 0.04 to detect the table with all objects and a threshold of 0.03 to detect the passthrough objects, which are the robot arm, robot gripper and the coke can or the blue can in this case. In experiments discussed in Sections 5.2 and 5.3, we use the threshold of 0.3 to detect the table or the open drawer where we want to add new distractors.

For generating LLM-assisted prompts, we perform 1-shot prompting to the LLM. For example, in the setting of generating novel distractors in the task where we place objects into the drawer (Section 5.2), we use the following prompt to the LLM:

```
Source task:  place pepsi can on the counter
Target task:  place pepsi can on the clutter counter
ViT region prompt:  empty counter
passthrough object prompt:  robot arm, robot gripper
inpainting prompt:  add a chip bag on the counter
Source task:  place coke can into top drawer
Target task:  place coke can into cluttered top drawer
```

and LLM generates the following prompt for detecting masks and augmentations (light blue means LLM generated):

```
ViT region prompt:  empty drawer
passthrough object prompt:  robot arm, robot gripper
inpainting prompt:  add a box of crackers in the drawer
```

which is semantically meaningful for performing mask detection and Imagen Editor augmentation. We follow this recipe of prompting for all of the tasks in our experiments.

During inpainting, we take the checkpoint of Imagen Editor 64x64 base model and the 256x256 super-resolution model trained in [59] and directly run inference to produce augmentations.

During evaluation, for the tasks that perform moving objects near novel containers and grasping unseen microfiber cloth, we perform 10 policy rollouts per new container/microfiber cloth of each method. For tasks that perform placing objects into novel containers, we perform 8 policy rollouts per new container for each method. For the task where the robot is instructed to place coke can or pepsi can into the unseen kitchen sink, for each method, we perform 5 policy rollouts for coke can and pepsi can respectively. For the task where the robot is instructed to grasp the coke can and the pepsi can in new backgrounds, we evaluate each method with 10 rollouts. For the task where the robot places the object into the cluttered drawer, we perform 10 policy rollouts per object for each method. Finally, for the task that requires the robot to pick up coke can in a scene with multiple coke cans, we perform 27 policy rollouts for each approach.

A.2 Computation Complexity

We train our policy on 16 TPUs for 1 day. For obtaining segmentation masks, we perform inference of OWL-ViT on 1 TPU for 1 hour to generate 1k episodes. During augmentation, we perform inference of Imagen Editor using 4 TPUs of the 64 x 64 base model and the 256 x 256 super-resolution model respectively for 2 hours to generate 1k episodes.

B Examples of Augmentations

We include more visualizations of augmentations generated by ROSIE in this section. In Figure 10, we show the generated episodes of ROSIE where we inpaint novel containers in the scene, which are used in the **Learning to move objects near generated novel containers** and **Learning to place objects into generated unseen containers** experiments in Section 5.1.

In Figure 8 and Figure 9, we visualize augmented episodes with new distractors, e.g. cluttered coke cans on the table and chip bags in the empty open drawer. These augmentations correspond experiments conducted in Section 5.2.

We also visualize the attention layers in RT-1 when training on our augmented data. As seen in Fig. 11, there are attention heads focusing on our augmented objects, which indicates the augmentation seem to be effective.

Overall, note that ROSIE is able generate semantically realistic novel objects and distractors in the manipulation setting. For example, ROSIE-generated objects typically has realistic shades on the table or the drawer, which is beneficial for training manipulation policies on top of such data.

Figure 8: Augmentation Example - adding a distractor can on the table.

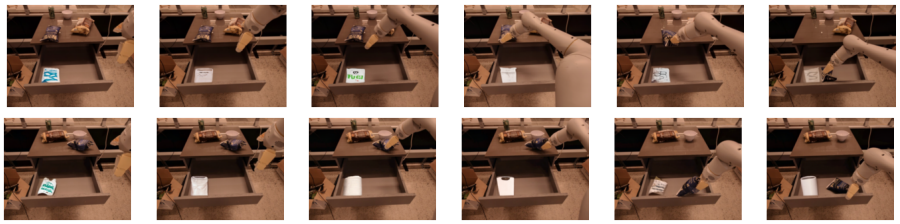
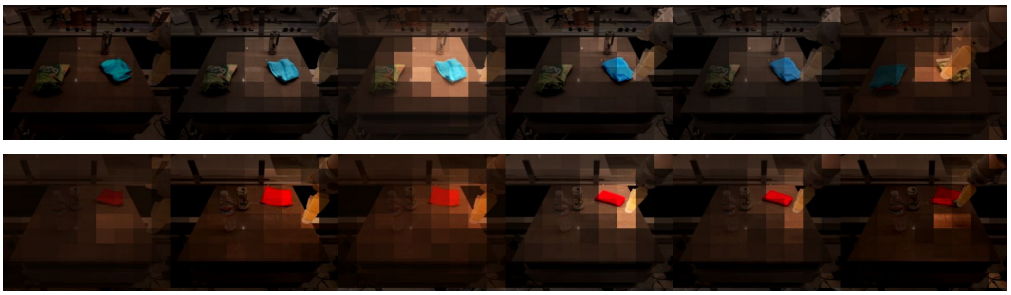


Figure 9: Augmentation Example - adding distractor objects into the drawer.

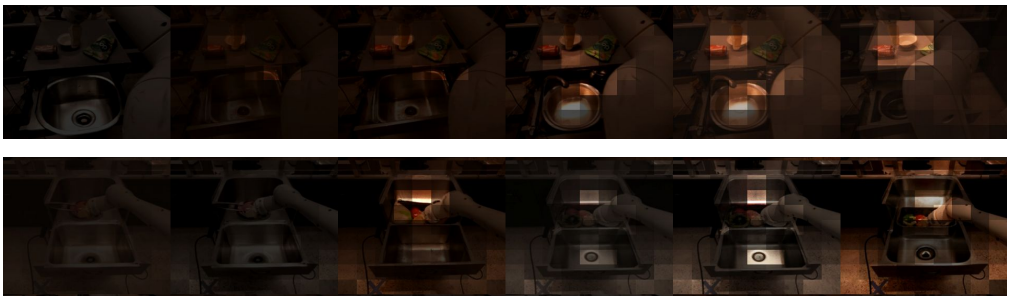
Figure 10: Augmentation Example - changing the container.

C Failure Cases of Generated Prompts and Images

While our LLM-assisted prompts generally work very well, we would like to note that it requires few-shot prompting to work well. In the zero-shot case, LLM would just hallucinate and output



pick blue/red microfiber cloth



place coke in sink

Figure 11: Visualization of some attention heads focusing on our augmented objects. This visualization is an overlay of observation and the spatial attention (bright regions means high attention).

unuseful augmentation prompts. For example, if we provide the following zero-shot prompt:

```
Source task:  pick coke can on a table
Target task:  pick coke can near a sink
Goal:  replace the scene in the source task with the scene in the target task
inpainting prompt:
```

and LLM gives the following response:

```
Pick up the coke can near the sink, replacing the one originally on the table
```

,which is not correct. Therefore few-shot prompting is crucial in ROSIE.

We show the failure cases of the augmented images in Figure 12. For the two examples on the left, ROSIE is supposed to generate woven basket and glass mason jar respectively, but it fails to generate such containers and instead generate some bowl-shape containers. For the two examples on the right, ROSIE is supposed to replace the in-hand green chip bag with blue microfiber cloth and a yellow rubber duck respectively. However, as the mask of the in-hand object becomes irregular, the performance of ROSIE degrades and ROSIE is unable to generate blue microfiber cloth and the yellow rubber duck in full shape and half of the in-hand object remains as the green chip bag. We suspect that with fine-tuning Imagen Editor on robotic datasets that show more manipulation-related data, we can improve the generation results drastically. Note that while the generation could be suboptimal at times, our insight is that such imperfect generation can only lead to misalignment between the task instruction and images, which may not have a big negative impact on the policy results and could give extra data augmentation benefit for free. Our policy performance in Section 5 validates this insight to some degree.

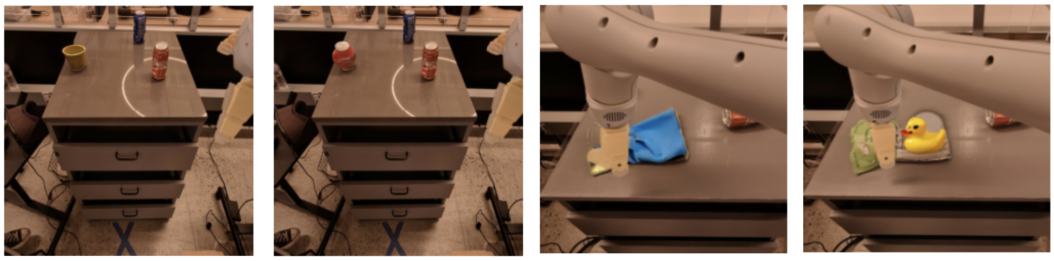


Figure 12: Failure cases of image augmentations.