Composable Part-Based Manipulation

Anonymous Author(s)
Affiliation
Address
email

Abstract: In this paper, we propose *composable part-based manipulation* (CPM), a novel approach that leverages object-part decomposition and part-part correspondences to improve learning and generalization of robotic manipulation skills. By considering the functional affordances between object parts, we conceptualize functional actions, such as pouring and constrained placing, as combinations of different constraints. CPM comprises a collection of composable diffusion models, where each model captures a different inter-object correspondence. These diffusion models can generate parameters for manipulation skills based on the specific object parts. Leveraging part-based correspondences coupled with the task decomposition into distinct constraints enables strong generalization to novel objects and object categories. We validate our approach in both simulated and real-world scenarios, demonstrating its effectiveness in achieving robust and generalized manipulation capabilities. Supplementary video can be found at sites.google.com/view/part-based-manipulation.

Keywords: Manipulation, Part Decomposition, Diffusion Model

1 Introduction

Compositionality provides appealing benefits in robotic manipulation, as it enables efficient learning, reasoning, and planning. Prior works have extensively studied the decomposition of scenes into objects and their relationships [1, 2, 3], as well as the division of long-horizon plans into primitive skills [3, 4], in order to navigate complex environments and devise long-horizon plans. In this paper, we present a different view of compositionality by considering *object-part decomposition* based on functionality (e.g., *rim, handle, body*), and leverage such decomposition to improve the learning of geometric and physical relationships for robot manipulation.

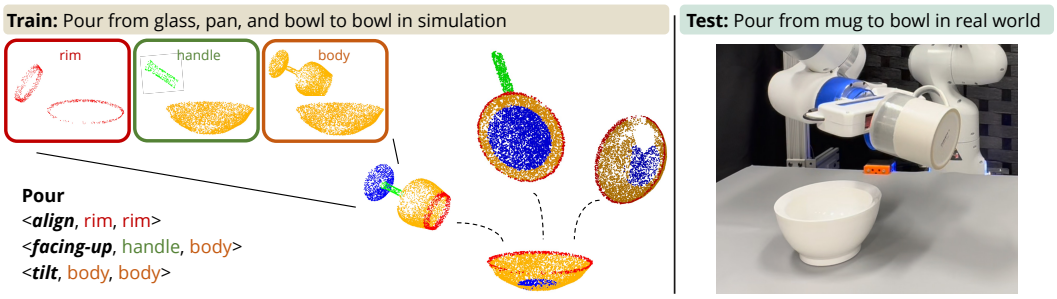

Figure 1: CPM composes part-based diffusion models to generalize manipulation across novel object categories, and from simulation to the real world.

In the context of language descriptions of objects, part names not only describe the geometric shapes of the parts but also capture their functional affordances. For instance, as depicted in Figure 1, for the action of "pouring", the *rims* define the boundary for alignment between the objects, the *bottom* of the pouring vessel should be tilted for the action, and its *handle* provides a constraint on the direction the object should face when pouring. Leveraging this knowledge of part affordances, we posit that

Submitted to the 7th Conference on Robot Learning (CoRL 2023). Do not distribute.

a family of functional actions, such as pouring and constrained placing, can be conceptualized as a combination of functional correspondences between object parts. Modeling actions using such a decomposition yields two important generalizations. First, it enables action generalization to novel instances from the same object category. Second and more importantly, it facilitates generalization to *unseen* object categories. For example, after learning part affordances for the "pouring" action, our robot trained on "pour from *bowls*" and "*...pans*" can generalize to "pour from *mugs*", with no additional training necessary for manipulation with the new object category.

Motivated by these insights, we present the *composable part-based manipulation* (CPM). CPM comprises a collection of diffusion models, where each model captures the functional correspondence between parts of different objects. These conditional diffusion models take the geometry of the object parts as input and generate parameters for manipulation skills, such as the starting and ending poses of a bowl during pouring actions. Specifically, the model outputs a distribution of feasible trajectories that satisfy particular correspondence constraints. After learning a collection of composable diffusion models for individual functional correspondences, we represent actions as combinations of part-part correspondences. During inference, we leverage the composition of primitive diffusion models to sample trajectories that adhere to all the part correspondence constraints. This approach improves generalization to novel object categories, including novel object parts, over models that do not reason about both parts and composable constraints.

In summary, this paper makes three key contributions. First, we propose the modeling of manipulation actions based on part-part correspondences between objects. Second, we present the composable part-based manipulation model, composed of diffusion models trained to capture primitive functional correspondences that can be flexibly recombined during inference. Third, CPM achieves strong generalization across various dimensions, including novel object instances and object categories. We validate the efficacy of CPM on both PyBullet-based simulations and real-robot experiments.

2 Related Work

Object representations for manipulation. Prior works use segmentations of common object parts (e.g., blades, lids, and handles) for manipulating articulated objects [5, 6, 7, 8] as well as for transfer to novel objects [9, 10]. A common approach that has been shown effective across different manipulation domains [11, 12, 13] first predicts which part of an object the robot should focus on (e.g., the handle), and then predicts an action relative to the part. Closely related is visual affordance detection [14, 15, 16], which segments objects into different functional regions, such as graspable parts and support surfaces of objects. These functional regions can be shared by more distinct objects, and have useful for generalizing task-oriented grasping between object categories [17, 18]. Keypoints are another representation that shows robustness to large intra-category shape variation and topology changes [19]. Each keypoint set can provide essential pose information, that lacks in previous segmentation approaches, to support tasks such as hanging mugs on pegs by their handles. The initial supervised approach [19] has been extended to methods that discover keypoints from interactions [20, 21] and from unlabeled videos [22]. Recently, implicit object representations have been used to provide correspondence between any point within the same object category generalizing across 6-DoF pose changes [23, 24, 25]. Large pretrained vision models also support the development of object representations; recent works leverage these models to significantly reduce domain-specific training data, showing strong results for open-vocabulary part segmentation [26], few-shot affordance segmentation [27], and one-shot pose estimation on any novel object from the same category [28]. Despite this huge progress, we still lack object representations that support strong generalization of manipulation between object categories. We focus on tackling this problem.

Learning interactions of objects. Works in robotics have established the importance of modeling interactions of objects. Recent approaches directly work on 3D observations, without relying on known object models. Learning spatial relations between objects enables the picking and placing of objects at specific locations [1, 29, 30, 2, 31], such as placing an object in the middle drawer, stacking objects, and placing multiple objects in a circle. These relations can be extended to represent the

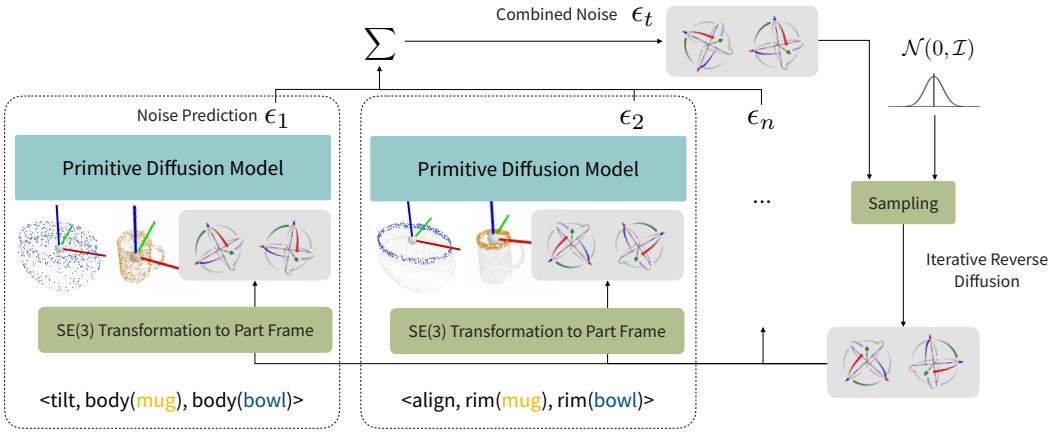

Figure 2: To sample the target poses (highlighted in grey) for a task l given point clouds of the function and anchor objects, CPM combines the noise prediction ϵ predicted by a collection of primitive diffusion models at inference time. During training, each diffusion model learns a target pose distribution that satisfies a particular part-part correspondence $c \in C_l$. Since each diffusion model is trained on the target pose distribution in its respective part frame, we transform the target pose to different part frames during inference.

logical state of the world to support planning for long-horizon tasks [3, 32, 33]. Other works focus on learning lower-level interactions between objects, such as placing an object stably on a messy tabletop and pushing an object using a tool [34, 35]. For example, O2O-afford [34] correlates feature maps extracted from two objects using a point convolution and outputs a point-wise interaction heatmap. Functionals defined on top of object-wise signed distance functions can also represent constraints on interactions between objects such as contact, containment, and task success [36]. Flow-based methods can also learn static relations between objects [37] as well as tool use [38], directly from point clouds. A main difference between our work and these methods is that we bridge the modeling of interactions and object representations through object-part decomposition and learned part-part correspondences, and enjoy empirically validated improvement in generalization.

Composable diffusion models. A set of recent works have investigated the potential of diffusion models in robotics [39, 40, 41, 42, 43, 44, 45, 46, 2, 47]. Research demonstrates that diffusion models can generate multimodal distributions over actions [41] and can handle spatial ambiguities in symmetric objects [2]. In image domains, prior work has shown a connection between conditional diffusion models and energy-based models, and proposed techniques to generate images by combining diffusion noises for different language conditions [48]. Recent work provides a more principled way to sample from individually trained models using MCMC [49]. Another approach combines diffusion models by using additional trained adapters for generating faces [50]. CPM combines both lines of work to propose composable diffusion models for robotic manipulation. In doing so we must address two challenges of adapting diffusion models to (1) output poses instead of pixels and (2) combine actions in different 6D coordinate frames, while retaining generalization to different distributions.

3 Composable Part-Based Manipulation

Composable part-based manipulation (CPM) comprises a collection of diffusion models, each defining a different type of part correspondence. These models can be separately trained and combined during inference to generate robot actions that align part correspondences. Formally, we are given a dataset $D = \{(x_a, x_f, T_f, l)\}_{i=1}^{M}$ of simulated robot task demonstrations. Here, x_a and x_f correspond to the point clouds of the static anchor object a and the function object f that is being manipulated, T_f is the target pose for the function object, and l is the name of manipulation task. We also assume access to a set of symbolic constraints C_l associated with each task l, which constrain the distribution of target poses for the task. We seek to learn a model of the form $T_f = g(x_a, x_f, l)$ to predict relevant manipulation trajectories for novel object point clouds, including objects from previously unseen categories. We introduce our composable diffusion models below.

3.1 Action as Part-Based Functional Correspondences

In this paper, we demonstrate CPM's capability to model interactions that involve two objects: an anchor and a function object. Consider the example of pouring from a mug to a bowl, as depicted in Fig. 1. This action involves three functional correspondences between parts. First, the *rim* of the mug should be aligned with the *rim* of the bowl. Second, the *body* of the mug should be tilted such that the contents can flow to the target bowl. Third, the *handle* of the mug should be facing up such that it does not obstruct the pouring action. The task of predicting robot motion can thus be cast as the task of finding a robot trajectory that simultaneously satisfies all these part-based functional correspondences.

Formally, each action l is associated with a fixed set of part-based functional correspondences C_l, which can also be viewed as part-based constraints over the robot motion. For example, as illustrated in Fig. 1, the action *pour* contains the following three correspondences: $\langle \textit{align}, \textit{rim}(\text{mug}), \textit{rim}(\text{bowl})\rangle$, $\langle \textit{tilt}, \textit{body}(\text{mug}), \textit{body}(\text{bowl})\rangle$, and $\langle \textit{facing-up}, \textit{handle}(\text{mug}), \textit{body}(\text{bowl})\rangle$. Here, each correspondence $c \in C$ is a tuple of $\langle \textit{type}(c), \textit{part}(f), \textit{part}(a)\rangle$, where *type* is the type of the correspondence, $\textit{part}(a)$ and $\textit{part}(f)$ refers to different parts of the anchor and the function object.

3.2 Composable Diffusion Models

In order to model individual part-based functional correspondences and compose them to predict robot motion, we employ diffusion models to learn a distribution of robot motion parameters based on part geometries. At a high-level, CPM contains a collection of primitive diffusion models. Each of them corresponds to a part-based correspondence type, e.g., *align* for rims. Throughout the paper, we parameterize the motion of the moving object (e.g., the mug) as two 6-DoF rigid form transformations, corresponding to the starting pose and the ending pose of the object movement. Fig. 1 shows the action of pouring water from a mug into a bowl. We start from a rest pose of the mug on the table and then move the mug to a location that is above the bowl and simultaneously tilt the mug.

Primitive diffusion models. In our implementation, each primitive diffusion model ϵ_θ takes in the current diffusion time step t, two object point clouds x_i and x_j, and the noisy starting and ending pose of the moving object T^{start} and T^{end} as input, and predicts the noise over T^{start} and T^{end}. As illustrated in Fig. 2, the denoise function is based on a transformer encoder. First, we encode point clouds for the two objects separately using a point cloud transformer. Then we encode the input poses using a trained MLP. We input the point cloud encodings and pose encodings, together with the diffusion time step t to the transformer encoder. The output of the transformer encoder is the predicted noise over two transformations T^{start} and T^{end}. During training, we optimize the following loss for randomly sampled diffusion time step t and random Gaussian noise ϵ sampled from a multivariate standard Gaussian distribution:

$$\mathcal{L}_{\text{MSE}} = \left\| \epsilon - \epsilon_\theta\left(\sqrt{1-\beta_t}T_0 + \sqrt{\beta_t}\epsilon \mid x_i, x_j, t\right)\right\|_2^2,$$

where $\mathcal{T}_0 = (T^{start}, T^{end})$ is the target poses to predict and β_t is the diffusion noise schedule [51].

Inference-time part frame-based pose representation. We make two optimizations to improve the performance of our composable diffusion models on the task of pose prediction. First, the added noise and the predicted noise are both in the tangent space of $SE(3)$. In practice, we first compute the exponential map of the transformations and then apply the noise. This can be viewed as predicting the score function for an exponential energy function of $SE(3)$ poses, similar to the $SE(3)$ Denoising Score Matching approach [40]. Second, we use a part-centric frame to represent both transformations. Intuitively, this corresponds to predicting the moving trajectory of the part (in the part frame) in contrast to the moving trajectory of the whole object (in the object frame). Previous work [52] has documented the advantage of this part frame-based representation in neural network pose prediction.

Inference-time combination. One of the key features of diffusion models is their compositionality. That is, suppose we have a set of diffusion models, each trained for one specific type of functional correspondences, we can combine their predicted noise during inference time to generate a trajectory that adheres to all functional correspondences. In particular, during inference time, starting from a

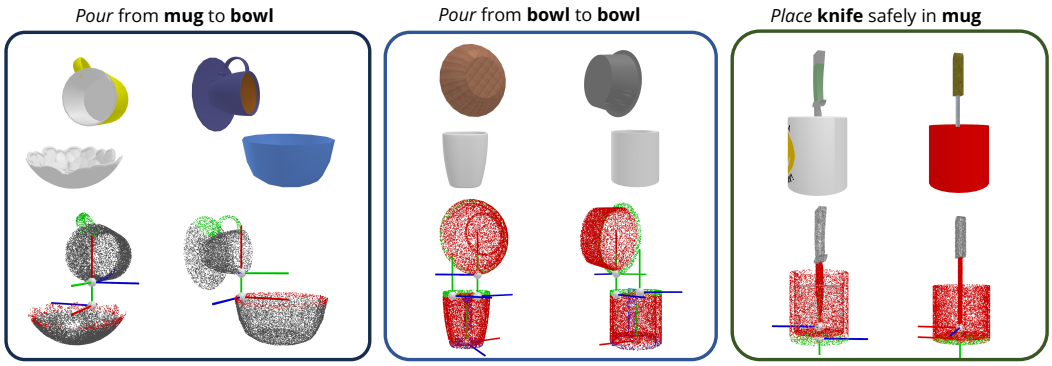

Figure 3: We generate object-centric task demonstrations using the PartNet and ShapeNetSem datasets for the "pouring" and "safe placing" tasks. For each task, we create demonstrations for a variety of function and anchor object combinations.

pair of poses $\mathcal{T}_K = (T^{start}, T^{end})$ randomly sampled from standard Gaussian distributions, given the set of constraints C_l associated with the action l, we iteratively update the pose prediction by:

$$\mathcal{T}_{t-1} = \frac{1}{\alpha_t}\left(\mathcal{T}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \sum_{c \in C_l} \epsilon_{type(c)}\left(\mathcal{T}_t \mid x_{part(a)}, x_{part(f)}, t\right)\right) + \sigma_t z,$$

where K is the number of diffusion steps, $\alpha_t = 1 - \beta_t$ is the denoise schedule, $\bar{\alpha}_t = \prod_{i=1}^{T} \alpha_t$ is the cumulated denoise schedule, σ_t is a fixed sampling-time noise schedule, and z is a randomly sampled Gaussian noise. In short, the key difference between our sampling process and a standard diffusion model sampling process is that we add together noises predicted by individual constraint models. Since each diffusion model implicitly parameterizes an energy-based model: $p_c(\mathcal{T}|cond) \propto \exp(-E_c(\mathcal{T}|cond))$ through its score [48, 49], this composition corresponding to maximize the probability of $\prod_{c \in C_l} p_c(\mathcal{T}|cond)$.

In the context of learning robot manipulation skills, our decomposition of part-based functional correspondences and the inference-time combination gives us better generalization to objects with similar part structures (e.g., cups and glasses); it also enables generalization to objects with different part structures (e.g., from mugs with handles to bowls without handles).

4 Data Collection

To validate our approach, we collect a dataset of task demonstrations, where each action l can be completed by using objects from different categories. Our data collection process involves sourcing 3D objects, extracting aligned parts, and rendering trajectories.

3D Objects. We source 3D objects based on a wide variety of 3D shapes from PartNet [53] and the subset of ShapeNetSem [54] objects categorized in the Acronym dataset [55]. We leverage 13 object categories to investigate manipulation generalization, including *mug*, *pan*, *bowl*, *wine glass*, *knife*, *can opener*, *scissors*, *screwdriver*, *fork*, *spoon*, *marker*, *pen*, and *flashlight*. Some object categories are reused for different tasks; for example, *mug* is used as an anchor for safe placing but also as an object for pouring.

Aligned Parts. Our method uses part segmentations of objects to learn primitive diffusion models. For 3D objects from PartNet, we use the segmentations provided in the dataset. For 3D objects from ShapeNetSem objects, we first label 3D keypoints, then from the labeled keypoints, we procedurally extract parts. As ShapeNet provides canonical poses for 3D models defined for each object category, we can also align the extracted functional parts for each object category.

Tasks. We demonstrate CPM on two tasks: "pouring" and "safe placing". These two tasks require different functional affordances. The pouring action pours from a function object to a target object, and requires alignment of *rims*, collision avoidance of *bottoms*, and *body* tilt. The safe-placing action

Table 1: CPM outperforms prior works in generalization to novel instances of objects within seen categories.

Model	Pouring	Safe Placing
PC-DDPM	75.83	51.55
Part-Aware PC-DDPM	75.28	42.68
CPM (ours)	**80.00**	**70.99**

Table 2: CPM demonstrates strong generalization to function objects from unseen object categories.

Model	Pouring				Safe Placing		
	Bowl	Glass	Mug	Pan	Fork	Pen	Scissors
PC-DDPM	63.02	75.95	71.39	64.39	40.60	46.63	32.34
Part-Aware PC-DDPM	58.98	72.11	67.11	**66.17**	39.76	48.04	28.15
CPM (ours)	**79.32**	**81.44**	**77.57**	62.13	**55.94**	**59.45**	**63.35**

places a sharp function object into a target object, and requires *head* containment for safety, *tip* touching *bottom*, and a *body-body* placement constraint. We procedurally generate parameters of the actions from the aligned parts. For each task, we also define model-based classifiers to verify task completion. For example, for safe placing, the tool end of the object (e.g., blade on a knife or tip of a screwdriver) needs to be contained by the body of the container.

Simulation and Rendering. We simulate the robot-object interactions by tracing the trajectories defined by the parameters. We first use multiple cameras to render RGB-D images, which yield realistic object point clouds. We then map the functional parts to the point clouds with the correct transformation and scaling. Finally, we obtain point cloud segments of each affordance part. Because these parts are extracted from the rendered point clouds, they can be incomplete and noisy, which increases the robustness of our method and helps transferability to real-world settings.

5 Experiments

The subsequent section will showcase the performance of CPM in comparison to baselines and other variants of our method based on a simulated environment. In particular, we evaluate in two important generalization settings: 1) generalization to novel object instances from seen object categories, and 2) generalizations to novel object instances from *unseen* object categories. We then discuss the deployment of CPM trained in simulation on a real robot.

5.1 Compared Methods

Baselines. We compare CPM with two main baselines. The first is PC-DDPM; similar to recent work [40, 47], we train a conditional denoising diffusion probabilistic model [51] that predicts target 6-DoF poses based on input point clouds of both the function and the anchor objects. This model encodes the point clouds of whole objects with a point cloud transformer [56]. The second baseline is the Part-Aware PC-DDPM, which also incorporates object part information akin to our CPM. In this model, part segmentation masks are provided as additional inputs for each point. The category label for each point is mapped to a learned embedding space before encoding.

CPM variants. We evaluate several variants of our model. The first is DDPM with 6D rotation representation instead of SE(3). This variant of CPM learns different diffusion models for different parts. However, it does not compose action in different coordinate frames. This model is directly adapted from existing composable diffusion models for image generation [48, 49]. The second is DDPM with training-time composition; this model composes energy functions from different part models at training time, in addition to inference-time composition. The last group are individual DDPMs, which use single DDPM models corresponding to different part-part correspondence in evaluation, without any composition.

5.2 Results

Comparisons to baselines. We evaluate CPM's generalization capability in two settings. First, Table 1 shows a comparison of generalization to novel objects from seen categories. Overall, our

model significantly outperforms baselines on both tasks of "pouring" and "safe placing". We note that although Part-Aware PC-DDPM leverages the same part segmentation as CPM, it fails to achieve stronger performance compared to the PC-DDPM baseline, which only uses the full point cloud as input. We attribute this to its potential overfitting to the part segmentations within the training data. By contrast, CPM is able to effectively leverage part segmentations by learning primitive diffusion models and composing them at inference time. Our model shows substantial improvements in the "safe placing" task, largely due to each part constraint significantly restricting the target pose distribution in this task. For instance, the constraint that requires the *tip* of the function object to touch the *bottom* of the anchor object effectively constrains the target pose.

Our second set of experiments assesses the model's capacity to generalize across *unseen* object categories, thereby highlighting the efficacy of learning part-based functional affordances. In this context, we evaluate models on function objects from categories not encountered during training. Results can be found in Table 2. Remarkably, CPM demonstrates its capability to generalize across object categories for both tasks in a zero-shot manner. However, CPM's performance dips slightly for pans as the *rim*'s of pans are significantly larger compared to *rim*'s encountered during training (for example, those of bowls and mugs). As a comparison, both PC-DDPM and Part-Aware PC-DDPM fall short in generalizing to new categories. Particularly noteworthy is CPM's performance on the "safe placing" task, where it surpasses the top-performing baseline by an average of 19.3%. This is largely attributable to the strong constraints imposed by part-based functional affordances in this task.

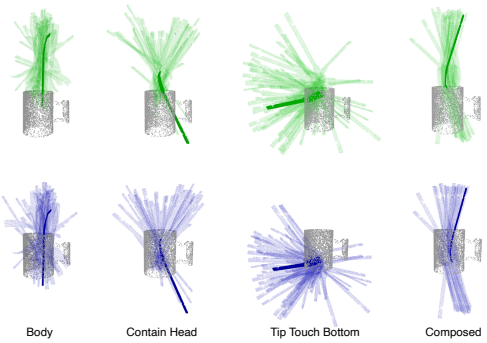

Figure 4: We qualitatively demonstrate the learned distribution of each individual diffusion model, which generates diverse samples conforming to the specified constraints, as well as the distribution from the combined full CPM model.

Ablation. First, we assess the significance of our SE(3) encoding, part frame-based transformation, and inference-time composition within the context of generalizing to unseen categories of objects. As depicted in Table 3, our full CPM with part frames and inference-time composition, shows superior performance compared to the model trained with training-time composition. This verifies the importance of our designs to support part-based composition and generalization. Compared to the variant based on 6D Rotation + 3D Translation encoding, CPM yields a better performance on the pouring task, a scenario where the rotation of the function object plays a pivotal role. On the safe placing task, which involves less rotation of objects, we observe a more comparable performance with our model. These results highlight the importance of SE(3) diffusion model in rotation prediction.

Second, we compare the performance of composed part-based diffusion models with the performance of individual diffusion models. Shown in Table 4, the composed model outperforms individual diffusion models, showing the efficacy of our composition paradigm. In addition, these results show the importance of different part-based constraints for the given tasks. In the "pouring" task, *align* and *tilt* strongly constrain the target pose for the function object, while for the "safe placing" task, *contain* and *place* constraints are more salient. Fig. 4 provides a qualitative visualization by showcasing the part-conditioned distribution associated with each individual diffusion model for various constraints, as well as the corresponding composed distribution. The quantitative performance of *contain* and *place* primitive models for these tasks aligns with this qualitative comparison, as they have learned distributions that are close to the composed model. The CPM paradigm allows us to train each primitive diffusion model independently, encouraging each model to concentrate on distinct functional affordances, thus enabling them to learn and generalize to diverse distributions of samples. During inference, the composition of distributions learned by individual models enables CPM to find solutions that satisfy all constraints.

Table 3: We ablate the contributions of CPM on the ability to generalize to novel categories of objects.

Target Pose Rep	Part Frames	Composition	Pouring	Safe Placing
6D Rot + 3D Trans	No	Inf-time	71.22	**68.77**
SE(3)	Yes	Train-time	69.89	48.46
SE(3)	Yes	Inf-time	**75.11**	59.58

Table 4: We explore the effect of composition, comparing to individual diffusion models, in generalization across both "pouring" and "safe placing" tasks. *We note that for the *align* and *facing-up* evaluation, a small percentage of examples were removed as they do not contain the involved parts in the partial object point clouds.

Pouring		Safe placing	
$\langle align, rim, rim \rangle$	70.05*	$\langle contain, head, body \rangle$	41.22
$\langle facing\text{-}up, handle, body \rangle$	16.42*	$\langle touch, tip, bottom \rangle$	9.34
$\langle tilt, body, body \rangle$	68.69	$\langle place, body, body \rangle$	39.86
CPM	**75.11**	CPM	**59.58**

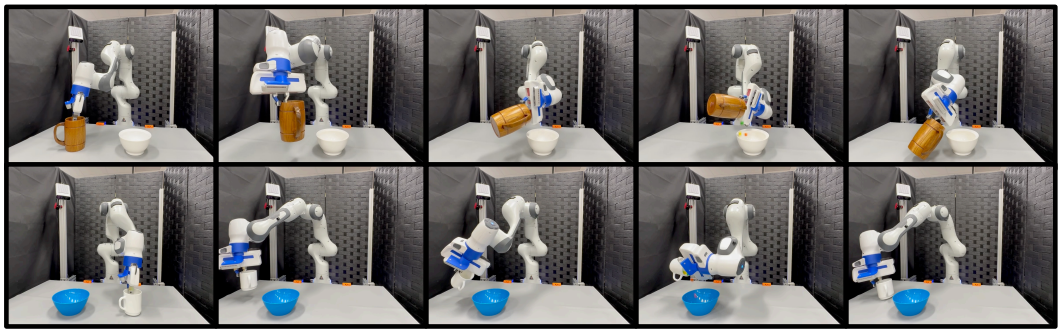

Figure 5: We show sampled frames from trajectories of CPM's policy in the real world, with a model trained from simulation data only.

Real-world transfer. Finally, we show a real-world robot manipulation experiment for the "pouring" task, highlighting the transferability of our CPM to real manipulation tasks. In this setting, we use the object trajectory generator learned by CPM in simulation with function objects of glass, pan, and bowl, and zero-shot transfer to mugs in the real-world experiment. Our setup includes the Franka Emika robot mounted in a tabletop environment with a mug and a bowl. To conduct pouring, we perform plane segmentation and k-means clustering to extract object point clouds from the point cloud captured by two calibrated Azure Kinect RGB-D cameras. Next, we apply a pre-trained point transformer (PT) model [57] for part segmentation. The segmentation model is trained on simulation data only. We then apply CPM trained in simulation to predict object poses for the pouring task. To execute the trajectory, we use contact-graspenet [58] to sample robot grasps on the function object and the Operational Space Controller [59] with impedance from Deoxys [59] to following a sequence of end-effector pose waypoints. Figure 5 shows our real-world setup and example trajectories predicted by CPM on unseen mugs with different shapes and sizes.

6 Limitations and Conclusion

We introduced composable part-based manipulation (CPM), as an approach that leverages object-part decomposition and part-part correspondences for robotic manipulation. We show that representing actions as combinations of constraints between object parts enables strong generalization. Through the composition of primitive diffusion models, we gain generalization capabilities across novel instances of objects as well as unseen object categories, in simulation and in real-world robot experiments.

Though CPM can be applied to unseen objects, a limitation is its reliance on object segmentation for perception and task-space control to execute actions. In addition, we do not currently differentiate constraints in different temporal windows of actions, which more complex tasks may require. Composing a larger number of models may require more efficient sampling techniques such as [60].

References

[1] C. Paxton, C. Xie, T. Hermans, and D. Fox. Predicting Stable Configurations for Semantic Placement of Novel Objects. In *Conference on Robot Learning (CoRL)*, 11 2021. URL https://arxiv.org/abs/2108.12062. 1, 2

[2] W. Liu, Y. Du, T. Hermans, S. Chernova, and C. Paxton. StructDiffusion: Language-Guided Creation of Physically-Valid Structures using Unseen Objects. In *Robotics: Science and Systems*, 2023. URL https://structdiffusion.github.io/. 1, 2, 3

[3] Y. Huang, A. Conkey, and T. Hermans. Planning for Multi-Object Manipulation with Graph Neural Network Relational Classifiers. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2023. URL https://arxiv.org/abs/2209.11943. 1, 3

[4] C. R. Garrett, R. Chitnis, R. Holladay, B. Kim, T. Silver, L. P. Kaelbling, and T. Lozano-Pérez. Integrated task and motion planning. *Annual review of control, robotics, and autonomous systems*, 4:265–293, 2021. URL https://arxiv.org/abs/2010.01083. 1

[5] 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. 2

[6] Z. Xu, Z. He, and S. Song. Universal manipulation policy network for articulated objects. *IEEE Robotics and Automation Letters*, 7(2):2447–2454, 2022. 2

[7] R. Wu, Y. Zhao, K. Mo, Z. Guo, Y. Wang, T. Wu, Q. Fan, X. Chen, L. Guibas, and H. Dong. Vat-mart: Learning visual action trajectory proposals for manipulating 3d articulated objects. *arXiv preprint arXiv:2106.14440*, 2021. 2

[8] H. Geng, H. Xu, C. Zhao, C. Xu, L. Yi, S. Huang, and H. Wang. Gapartnet: Cross-category domain-generalizable object perception and manipulation via generalizable and actionable parts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7081–7091, 2023. 2

[9] J. Aleotti and S. Caselli. Manipulation planning of similar objects by part correspondence. In *International Conference on Advanced Robotics (ICAR)*, pages 247–252, 2011. doi:10.1109/ICAR.2011.6088558. 2

[10] N. Vahrenkamp, L. Westkamp, N. Yamanobe, E. E. Aksoy, and T. Asfour. Part-based grasp planning for familiar objects. In *IEEE-RAS international conference on humanoid robots (humanoids)*, pages 919–925. IEEE, 2016. 2

[11] P. Parashar, J. Vakil, S. Powers, and C. Paxton. Spatial-language attention policies for efficient robot learning. *arXiv preprint arXiv:2304.11235*, 2023. 2

[12] O. Mees, J. Borja-Diaz, and W. Burgard. Grounding language with visual affordances over unstructured data. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, London, UK, 2023. 2

[13] E. Valassakis, G. Papagiannis, N. Di Palo, and E. Johns. Demonstrate once, imitate immediately (dome): Learning visual servoing for one-shot imitation learning. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8614–8621. IEEE, 2022. 2

[14] A. Myers, C. L. Teo, C. Fermüller, and Y. Aloimonos. Affordance detection of tool parts from geometric features. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1374–1381. IEEE, 2015. 2

[15] T.-T. Do, A. Nguyen, and I. Reid. Affordancenet: An end-to-end deep learning approach for object affordance detection. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 5882–5889. IEEE, 2018. 2

[16] S. Deng, X. Xu, C. Wu, K. Chen, and K. Jia. 3d affordancenet: A benchmark for visual object affordance understanding. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1778–1787, 2021. 2

[17] W. Liu, A. Daruna, and S. Chernova. Cage: Context-aware grasping engine. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2550–2556. IEEE, 2020. 2

[18] P. Ardón, È. Pairet, R. P. Petrick, S. Ramamoorthy, and K. S. Lohan. Learning grasp affordance reasoning through semantic relations. *IEEE Robotics and Automation Letters*, 4(4):4571–4578, 2019. 2

[19] L. Manuelli, W. Gao, P. Florence, and R. Tedrake. kpam: Keypoint affordances for category-level robotic manipulation. In *Robotics Research: The 19th International Symposium ISRR*, pages 132–157. Springer, 2022. 2

[20] Z. Qin, K. Fang, Y. Zhu, L. Fei-Fei, and S. Savarese. Keto: Learning keypoint representations for tool manipulation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7278–7285. IEEE, 2020. 2

[21] D. Turpin, L. Wang, S. Tsogkas, S. Dickinson, and A. Garg. Gift: Generalizable interaction-aware functional tool affordances without labels. *arXiv preprint arXiv:2106.14973*, 2021. 2

[22] L. Manuelli, Y. Li, P. Florence, and R. Tedrake. Keypoints into the future: Self-supervised correspondence in model-based reinforcement learning. *arXiv preprint arXiv:2009.05085*, 2020. 2

[23] A. Simeonov, Y. Du, A. Tagliasacchi, J. B. Tenenbaum, A. Rodriguez, P. Agrawal, and V. Sitzmann. Neural descriptor fields: Se (3)-equivariant object representations for manipulation. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 6394–6400. IEEE, 2022. 2

[24] E. Chun, Y. Du, A. Simeonov, T. Lozano-Perez, and L. Kaelbling. Local neural descriptor fields: Locally conditioned object representations for manipulation. *arXiv preprint arXiv:2302.03573*, 2023. 2

[25] J.-S. Ha, D. Driess, and M. Toussaint. Deep visual constraints: Neural implicit models for manipulation planning from visual input. *IEEE Robotics and Automation Letters*, 7(4):10857–10864, 2022. 2

[26] M. Liu, Y. Zhu, H. Cai, S. Han, Z. Ling, F. Porikli, and H. Su. Partslip: Low-shot part segmentation for 3d point clouds via pretrained image-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21736–21746, 2023. 2

[27] D. Hadjivelichkov, S. Zwane, L. Agapito, M. P. Deisenroth, and D. Kanoulas. One-shot transfer of affordance regions? affcorrs! In *Conference on Robot Learning*, pages 550–560. PMLR, 2023. 2

[28] W. Goodwin, I. Havoutis, and I. Posner. You only look at one: Category-level object representations for pose estimation from a single example. *arXiv preprint arXiv:2305.12626*, 2023. 2

[29] W. Yuan, C. Paxton, K. Desingh, and D. Fox. Sornet: Spatial object-centric representations for sequential manipulation. In *Conference on Robot Learning*, pages 148–157. PMLR, 2022. 2

[30] W. Liu, C. Paxton, T. Hermans, and D. Fox. StructFormer: Learning Spatial Structure for Language-Guided Semantic Rearrangement of Novel Objects. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2022. URL https://sites.google.com/view/structformer. 2

[31] M. Shridhar, L. Manuelli, and D. Fox. Cliport: What and where pathways for robotic manipulation. In *Conference on Robot Learning*, pages 894–906. PMLR, 2022. 2

[32] T. Silver, R. Chitnis, J. Tenenbaum, L. P. Kaelbling, and T. Lozano-Pérez. Learning symbolic operators for task and motion planning. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3182–3189. IEEE, 2021. 3

[33] K. Kase, C. Paxton, H. Mazhar, T. Ogata, and D. Fox. Transferable task execution from pixels through deep planning domain learning. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10459–10465. IEEE, 2020. 3

[34] K. Mo, Y. Qin, F. Xiang, H. Su, and L. Guibas. O2o-afford: Annotation-free large-scale object-object affordance learning. In *Conference on Robot Learning*, pages 1666–1677. PMLR, 2022. 3

[35] J. Liang and A. Boularias. Learning category-level manipulation tasks from point clouds with dynamic graph cnns. *arXiv preprint arXiv:2209.06331*, 2022. 3

[36] D. Driess, J.-S. Ha, M. Toussaint, and R. Tedrake. Learning models as functionals of signed-distance fields for manipulation planning. In *Conference on Robot Learning*, pages 245–255. PMLR, 2022. 3

[37] C. Pan, B. Okorn, H. Zhang, B. Eisner, and D. Held. Tax-pose: Task-specific cross-pose estimation for robot manipulation. In *Conference on Robot Learning*, pages 1783–1792. PMLR, 2023. 3

[38] D. Seita, Y. Wang, S. J. Shetty, E. Y. Li, Z. Erickson, and D. Held. Toolflownet: Robotic manipulation with tools via predicting tool flow from point clouds. In *Conference on Robot Learning*, pages 1038–1049. PMLR, 2023. 3

[39] M. Janner, Y. Du, J. Tenenbaum, and S. Levine. Planning with diffusion for flexible behavior synthesis. In *ICML*, 2022. 3

[40] J. Urain, N. Funk, J. Peters, and G. Chalvatzaki. Se(3)-diffusionfields: Learning smooth cost functions for joint grasp and motion optimization through diffusion. *IEEE International Conference on Robotics and Automation (ICRA)*, 2023. 3, 4, 6

[41] S. Huang, Z. Wang, P. Li, B. Jia, T. Liu, Y. Zhu, W. Liang, and S.-C. Zhu. Diffusion-based generation, optimization, and planning in 3d scenes. In *CVPR*, 2023. 3

[42] I. Kapelyukh, V. Vosylius, and E. Johns. Dall-e-bot: Introducing web-scale diffusion models to robotics. *IEEE Robotics and Automation Letters (RA-L)*, 2023. 3

[43] A. Ajay, Y. Du, A. Gupta, J. Tenenbaum, T. Jaakkola, and P. Agrawal. Is conditional generative modeling all you need for decision-making? *arXiv*, 2022. 3

[44] 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. 3

[45] U. A. Mishra and Y. Chen. Reorientdiff: Diffusion model based reorientation for object manipulation. *arXiv preprint arXiv:2303.12700*, 2023. 3

[46] C. Higuera, B. Boots, and M. Mukadam. Learning to read braille: Bridging the tactile reality gap with diffusion models. *arXiv preprint arXiv:2304.01182*, 2023. 3

[47] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Proceedings of Robotics: Science and Systems (RSS)*, 2023. 3, 6

[48] N. Liu, S. Li, Y. Du, A. Torralba, and J. B. Tenenbaum. Compositional visual generation with composable diffusion models. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XVII*, pages 423–439. Springer, 2022. 3, 5, 6

[49] Y. Du, C. Durkan, R. Strudel, J. B. Tenenbaum, S. Dieleman, R. Fergus, J. Sohl-Dickstein, A. Doucet, and W. Grathwohl. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and mcmc. *arXiv preprint arXiv:2302.11552*, 2023. 3, 5, 6

[50] Z. Huang, K. C. Chan, Y. Jiang, and Z. Liu. Collaborative diffusion for multi-modal face generation and editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6080–6090, 2023. 3

[51] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 4, 6

[52] M. Liu, X. Li, Z. Ling, Y. Li, and H. Su. Frame mining: a free lunch for learning robotic manipulation from 3d point clouds. In *6th Annual Conference on Robot Learning*, 2022. 4

[53] K. Mo, S. Zhu, A. X. Chang, L. Yi, S. Tripathi, L. J. Guibas, and H. Su. Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 909–918, 2019. 5

[54] M. Savva, A. X. Chang, and P. Hanrahan. Semantically-enriched 3d models for common-sense knowledge. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 24–31, 2015. 5

[55] C. Eppner, A. Mousavian, and D. Fox. Acronym: A large-scale grasp dataset based on simulation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6222–6227. IEEE, 2021. 5

[56] M.-H. Guo, J.-X. Cai, Z.-N. Liu, T.-J. Mu, R. R. Martin, and S.-M. Hu. Pct: Point cloud transformer. *Computational Visual Media*, 7:187–199, 2021. 6

[57] H. Zhao, L. Jiang, J. Jia, P. H. Torr, and V. Koltun. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16259–16268, 2021. 8

[58] M. Sundermeyer, A. Mousavian, R. Triebel, and D. Fox. Contact-graspnet: Efficient 6-dof grasp generation in cluttered scenes. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13438–13444. IEEE, 2021. 8

[59] Y. Zhu, A. Joshi, P. Stone, and Y. Zhu. Viola: Imitation learning for vision-based manipulation with object proposal priors. *arXiv preprint arXiv:2210.11339*, 2022. 8

[60] Q. Zhang and Y. Chen. Fast sampling of diffusion models with exponential integrator. *arXiv preprint arXiv:2204.13902*, 2022. 8