ReorientDiff: Diffusion Model based Reorientation for Object Manipulation

Utkarsh A. Mishra and Yongxin Chen
Institute for Robotics and Intelligent Machines (IRIM),
Georgia Institute of Technology
umishra31@gatech.edu, yongchen@gatech.edu

***Abstract*—The ability to manipulate objects in a desired configurations is a fundamental requirement for robots to complete various practical applications. While certain goals can be achieved by picking and placing the objects of interest directly, object reorientation is needed for precise placement in most of the tasks. In such scenarios, the object must be reoriented and re-positioned into intermediate poses that facilitate accurate placement at the target pose. To this end, we propose a reorientation planning method, ReorientDiff, that utilizes a diffusion model-based approach. The proposed method employs both visual inputs from the scene, and goal-specific language prompts to plan intermediate reorientation poses. Specifically, the scene and language-task information are mapped into a joint scene-task representation feature space, which is subsequently leveraged to condition the diffusion model. The diffusion model samples intermediate poses based on the representation using classifier-free guidance and then uses gradients of learned feasibility-score models for implicit iterative pose-refinement. The proposed method is evaluated using a set of YCB-objects and a suction gripper, demonstrating a success rate of 96.5% in simulation. Overall, our study presents a promising approach to address the reorientation challenge in manipulation by learning a conditional distribution, which is an effective way to move towards more generalizable object manipulation. For more results, checkout our website: https://utkarshmishra04.github.io/ReorientDiff.**

I. Introduction

Rearranging objects in a desired pose is an important skill necessary for daily activities at home as well as for specific arrangement and packing applications in the industry. Performing such a task requires extracting object information from visual-sensor data and planning a pick-place sequence [20, 17]. While a single-step pick-place sequence is a viable solution, placing the object at a specific position and orientation is not always feasible. Reorientation is a helpful strategy when successfully changing an object's pose allows its placement at the target pose [19]. Reorientation ensures feasible intermediate transition poses in scenarios where there are no common grasps between the current pose and an object's desired placement pose.

Recently, Wada *et al.* [19] proposed a data-driven sampling-based solution for such a problem and performed reorientation using learned models that predict the feasibility score of an intermediate pose. While their method significantly improved the success rate and planning time, the approach relied on the target object's specification and placement pose. Lately, with the advances in language descriptor foundation models like CLIP [11], which projects images and texts to a common feature space, such specifications can be directly correlated between visual information and suitable language commands, thus empowering human-robot interaction. This motivated us to explore grounding the problem statement of reorientation on language and hence embed semantic knowledge of the task with the spatial structure of the scene [12].

In this paper, we propose ReorientDiff, a diffusion model-based reorientation pose generation pipeline for solving the task proposed by [19] for picking objects from a cluttered pile and placing it in the target pose specified through language descriptions. The core idea of our approach is to visualize the feasible intermediate poses as distributions. Such a distribution can be captured by a diffusion model and will be conditioned on the object's current and target pose, or in a more general multi-object scenario, on the pile of pickable objects and the occupancy of the target location. Note that diffusion models have also been successfully used for motion planning [7, 1], grasp planning [18], and object rearrangement [8] applications.

To enable interaction using natural language directly, we use pre-trained CLIP embeddings with an object segmentation model to structure object selection, pose prediction, and target object segmentation networks for the task. Considering the intermediate features as a generic scene and target representation in reduced dimensionality, the diffusion model samples reorientation poses conditioned on such features, which are further implicitly refined by a feasibility-score-based discriminator similar to the models used by [10, 19]. We combine a generic classifier-free conditional sampling [5] with classifier-guided sampling [4] to sample from diffusion models. For the tasks, we consider reorientation of objects in the YCB dataset [2] that are feasible for suction grippers. For each selected object, we choose target locations on multiple shelf levels and four possible target orientations. Our method samples reorientation poses in continuous space without any discretization or candidate pose selection and reaches a cumulative success rate of 96.5% as evaluated on selected individual objects.

II. Reorientation

Following the previous environment setup by Wada *et al.* [19], we construct the reorientation scenario as a task of i) selecting an object of interest from a pile of cluttered objects, ii) calculating feasible grasp poses for picking, iii) calculating grasp poses for placement with prior knowledge of the mesh of the selected object and iv) finding suitable reorientation poses

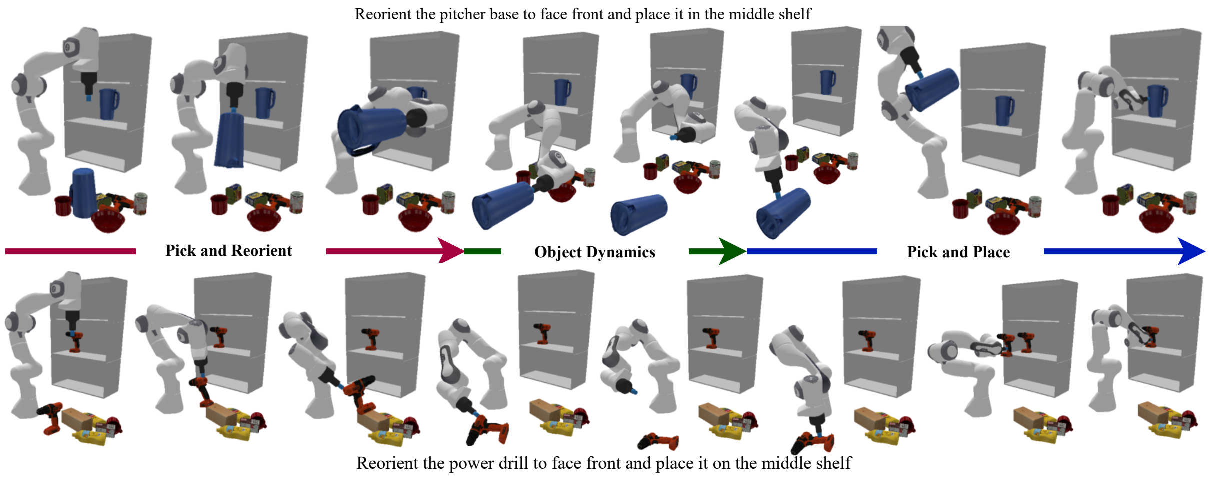

Reorient the pitcher base to face front and place it in the middle shelf
Pick and Reorient
Object Dynamics
Pick and Place
Reorient the power drill to face front and place it on the middle shelf

Fig. 1: **Reorientation for precise target placement** The above figure represents the phenomenon of reorientation in which an object from a cluttered file has to be placed precisely in a shelf (target position shown). As the object cannot be directly placed at the target location, our proposed method, ReorientDiff, samples a reorientation pose using a learned conditional distribution by a diffusion model. Such a proposed reorientation pose acts as a transition for facilitating successful placement. We also consider and take advantage of the object dynamics, as introduced by Wada *et al.* [19], by which we ensure that un-grasping an object in an unstable pose will eventually allow the object to settle at some favourable pose.

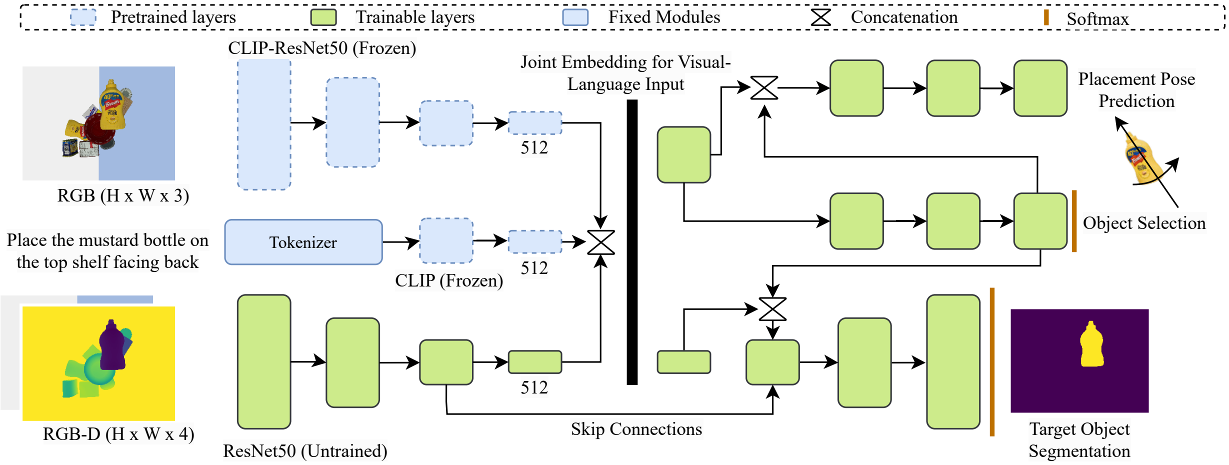


Fig. 2: **Joint Embedding Construction** We use a pre-trained CLIP-ResNet50 and BPE-based Tokenizer with CLIP language model for obtaining a semantic embedding of the tabletop RGB image and instruction prompt, respectively. While keeping CLIP layers frozen, we train another ResNet50 encoder for spatial RGB-D observations and combine them with the semantic embeddings to obtain joint embeddings for visual-language inputs. We train these latent representations with respect to the object of interest, placement pose, and object location (segmentation) predictions. It is worth noting that the predicted object information is also used for predicting the placement pose and the target object segmentation. Further, the addition of skip-connection also ensures that the segmentation map construction is accurate while filling up the embedding vector with only the necessary information. The proposed pipeline shown above creates a latent space that is consistent with the three aspects of interest by minimizing information loss.

using our proposed pipeline based on diffusion models. This section describes the pipeline for creating a generic scene and task embedding space, followed by generating grasp poses and training the feasibility score models.

A. Constructing Generic Scene-Task Representations

We define a scene as the location and occupancy of the place from where a target object should be picked and a task as the language prompt containing the descriptions for selecting the target object and deciding placement poses. A top-down RGB-D camera provides an image $\mathcal{I} \in \mathbb{R}^{H \times W \times 3}$ and a heightmap $\mathcal{H} \in \mathbb{R}^{H \times W \times 1}$ as the description of the pile. Motivated from previous work [12, 13] on learning semantic and spatial embeddings, we use pre-trained CLIP foundation model for obtaining semantic embeddings from the image $\mathcal{I}$ and language $\mathcal{L}$, and combine them with spatial embeddings for target object segmentation to get a joint embedding Φ as generic scene-task representation as shown in Fig. 2. The embedding is further used to predict the target object as a one-hot vector of all the objects of interest and the final placement pose.

B. Sampling Grasp Poses

We generate grasp poses by following the classical approach of converting the heightmap into a point cloud representation and eventually to a point-normal representation. The predicted target object segmenatation of the scene is then used to obtain the surface normals of the target object. After performing an edge masking using the Laplacian of the surface normals, the remaining point-normals on the surface are feasible grasp poses.

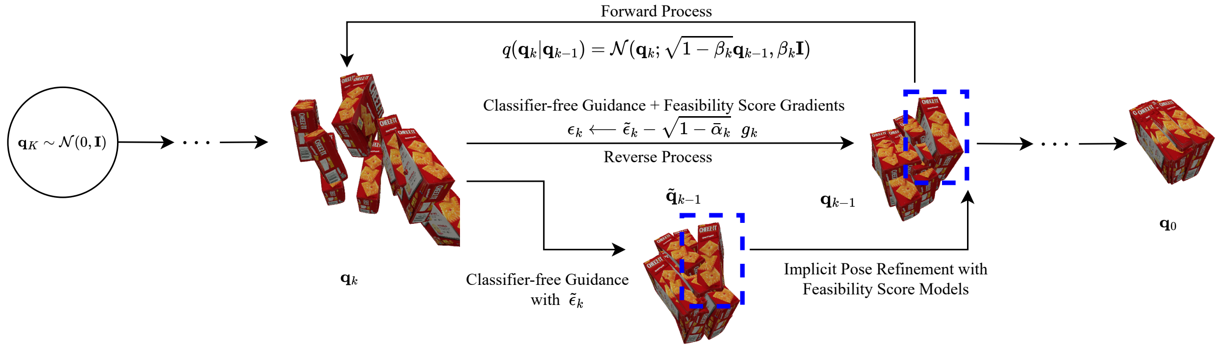

Forward Process
$q(\mathbf{q}_k|\mathbf{q}_{k-1}) = \mathcal{N}(\mathbf{q}_k; \sqrt{1-\beta_k}\mathbf{q}_{k-1}, \beta_k \mathbf{I})$
Classifier-free Guidance + Feasibility Score Gradients
$\epsilon_k \longleftarrow \tilde{\epsilon}_k - \sqrt{1-\bar{\alpha}_k}\ g_k$
Reverse Process
$\mathbf{q}_K \sim \mathcal{N}(0, \mathbf{I})$
$\mathbf{q}_k$
$\tilde{\mathbf{q}}_{k-1}$
$\mathbf{q}_{k-1}$
$\mathbf{q}_0$
Classifier-free Guidance with $\tilde{\epsilon}_k$
Implicit Pose Refinement with Feasibility Score Models

Fig. 3: **Forward and Reverse Diffusion Process** The above figure shows the forward diffusion and the reverse denoising and sampling process of ReorientDiff. As described in Section III, following classifier-free guidance will result in high-likelihood samples with high-variance in terms of success feasibility of the samples. Using the feasibility score gradients, we realize an implicit iterative pose refinement, as marked by the blue box in the figure. This significantly decrease variance and ensure high success feasibility of the samples.

While we sample grasp poses η_1 for picking the object from the pile in the aforementioned manner, we assume that we have the mesh of the selected object for sampling grasp poses η_2 for placing the object at the predicted pose.

C. Training Feasibility Score Models

Following prior works [10, 19, 9], a feasibility prediction model is important for early-evaluation and rejection of unfavourable samples. Such a feasibility model predicts the probability of success of a given grasp pose in successfully grasping an object in some candidate pose for a specified scene representation. The phenomenon of grasp success evaluation in dynamic reorientation pose, as addressed by [19], is of particular interest for our setup. Modelling dynamics for every object is indeed non-trivial and adds to the complexity; hence the feasibility model implicitly takes care of the dynamics of the object after deactivating the grasp. For checking feasibility or the probability of success (y) of sampled grasps for candidate reorientation poses $\mathbf{q}$, we train two models:

- For predicting success of reorientation from the current pose in a pile to a candidate pose given pick grasp poses (η_1) and scene representation, denoted as $\mathcal{M}_1(y|\eta_1, \mathbf{q}, \Phi)$
- For predicting success of post-grasp deactivation pose from the candidate pose and placement grasp poses (η_2), denoted as $\mathcal{M}_2(y|\eta_2, \mathbf{q}, \Phi)$

III. ReorientDiff: Diffusion for Reorientation

We aim to generate intermediate reorientation poses for the target object, which enables successive placement at the desired pose and is reachable from the current pose. ho We introduce a diffusion model based approach to sample most probable successful reorientation poses ($\mathbf{q}$) conditioned on the scene representation priors (Φ), denoted as $p(\mathbf{q}|\Phi)$, which already contains the spatial and semantic information about the scene and the task. The denoising process can be further flexibly conditioned by sampling from modified distributions of the form

$$p_h(\mathbf{q}) \propto p(\mathbf{q}|\Phi)h(\mathbf{q}, \Phi), \tag{1}$$

where $h(\mathbf{q}, \Phi)$ can represent several grasp success probability heuristics. By separating the grasp success from reorientation candidate sampling, the diffusion model trained for reorientation poses can be reused for varied selection of picking (η_1) and placement grasp poses (η_2).

A. Classifier-free Conditional Pose Generation

Following the distribution defined in (1), we use classifier-free guidance [5] to sample high-likelihood reorientation poses for a particular scene-task representation. We train a score-network [15], $\epsilon_\theta(\mathbf{q}_k, \Phi) \propto \nabla_{\mathbf{q}_k} \log p(\mathbf{q}_k|\Phi)$, to denoise from $\mathbf{q}_K \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ to possible reorientation poses $\mathbf{q}_0$ from a K-step reverse diffusion denoising process. For each step, we calculate $\tilde{\epsilon}_k$ as

$$\tilde{\epsilon}_k = \epsilon_\theta(\mathbf{q}_k, \Phi) + w_c\Big(\epsilon_\theta(\mathbf{q}_k, \Phi) - \epsilon_\theta(\mathbf{q}_k, \emptyset)\Big) \tag{2}$$

The scalar w_c implicitly guides the reverse-diffusion towards poses that best satisfy the scene-task representations. Further, we calculate the successive samples for the next $(k-1)^{th}$ step using the DDIM [14] sampling strategy and $\tilde{\epsilon}_k$ as follows:

$$\tilde{\mathbf{q}}_{k-1} \longleftarrow \sqrt{\bar{\alpha}_{k-1}}\Big(\frac{\mathbf{q}_k - \sqrt{1-\bar{\alpha}_k}\ \ \tilde{\epsilon}_k}{\sqrt{\bar{\alpha}_k}}\Big) + \sqrt{1-\bar{\alpha}_{k-1}}\ \ \tilde{\epsilon}_k \tag{3}$$

where, $\bar{\alpha}_k$ is as described in Section VI.

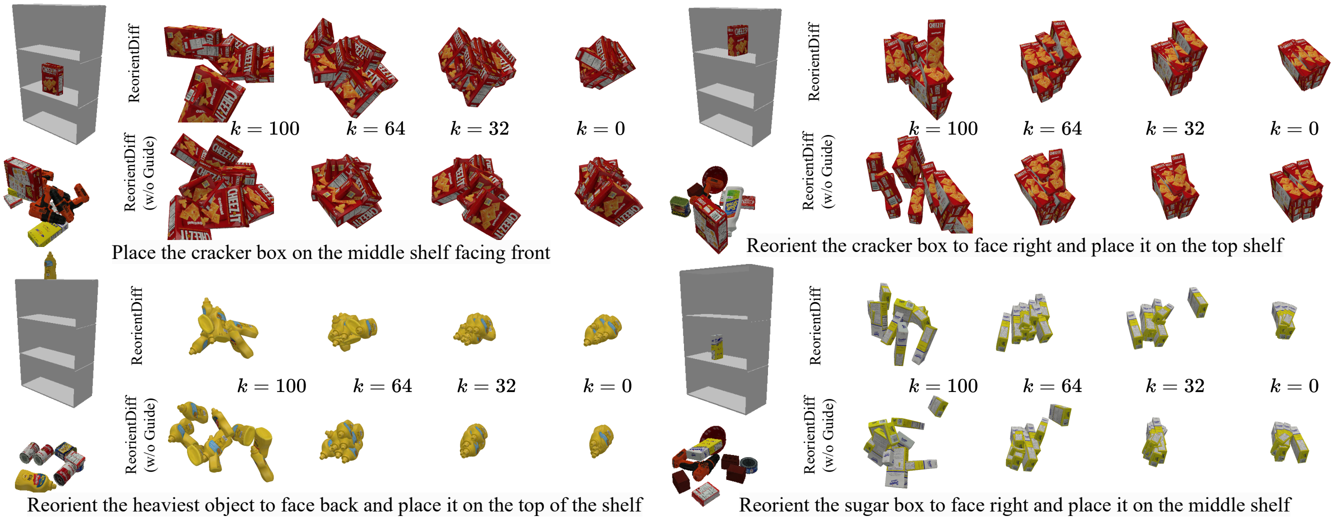
B. Feasibility Guided Pose Refinement

We use the two feasibility-score prediction models ($\mathcal{M}_1$ and $\mathcal{M}_2$), which are pre-trained for predicting grasp feasibility for picking grasp, reorientation pose pairs and placement grasp, reorientation pose pairs, respectively. In such a case, the scores can be converted into probability distributions for each heuristic, defined as, for each $i = 1, 2$,

$$h_i \equiv\ p(y=1|\eta_i, \mathbf{q}, \Phi)|_{\mathcal{M}_i} = \exp\Big(-(1-\mathcal{M}_i(y|\eta_i, \mathbf{q}, \Phi))^2\Big)$$

Following classifier-based guidance [4] formulation for the heuristics, the reverse diffusion can be formulated as:

$$p_h(\mathbf{q}_k|\mathbf{q}_{k+1}, y, \Phi) \propto \\ p(\mathbf{q}_k|\mathbf{q}_{k+1}, \Phi)\ \ p(y|\eta_1, \hat{\mathbf{q}}_0^k, \Phi)|_{\mathcal{M}_1}\ p(y|\eta_2, \hat{\mathbf{q}}_0^k, \Phi)|_{\mathcal{M}_2} \tag{4}$$



Place the cracker box on the middle shelf facing front

Reorient the cracker box to face right and place it on the top shelf

Reorient the heaviest object to face back and place it on the top of the shelf

Reorient the sugar box to face right and place it on the middle shelf

Fig. 4: **Reverse Diffusion for Reorientation Pose Generation** The reverse sampling process for 4 k-steps at $k = 100, 64, 32, 0$ for $K = 256$ in four different scene-task scenarios comprising of the Cracker Box, Mustard Bottle and Sugar Box in different target orientations are shown above. The scenes are shown in the left-side of every sub-figure and consists of the pile with the target object and the predicted placement location on the shelf. The language prompt defining each of the tasks is mentioned below each sub-figure. It consists of either the absolute (the object's name) or the relative (heaviest/lightest) reference to the object and details about the target placement.

where, $\hat{\mathbf{q}}_0^k$ is the sample proposed at diffusion step k and defined as:

$$\hat{\mathbf{q}}_0^k = \frac{\mathbf{q}_k - \sqrt{1-\bar{\alpha}_k}\ \ \tilde{\epsilon}_k}{\sqrt{\bar{\alpha}_k}} \tag{5}$$

Considering Taylor first order approximations for heuristics and standard reverse process Gaussian $(\mu_\theta(\mathbf{q}_k, k, \Phi), \beta_k \mathbf{I})$ as described in Section VI, we get the new mean $(\mu_{\theta,h}(\mathbf{q}_k, k, \Phi))$ for the distribution $p_h(\mathbf{q}_k|\mathbf{q}_{k+1}, y, \Phi)$ in (4) as:

$$\begin{aligned}
&\mu_{\theta,h}(\mathbf{q}_k, k, \Phi) \\
&= \mu_\theta(\mathbf{q}_k, k, \Phi) + \beta_k \sum_{i=1}^{2} w_i \nabla_{\mathbf{q}_k} \log p(y|\eta_i, \mathbf{q}_k, \Phi)|_{\mathcal{M}_i} \\
&= \mu_\theta(\mathbf{q}_k, k, \Phi) - \beta_k \sum_{i=1}^{2} w_i \nabla_{\mathbf{q}_k} \Big[1 - \mathcal{M}_i(y|\eta_i, \hat{\mathbf{q}}_0^k, \Phi)\Big]^2.
\end{aligned}$$

In view of (7), we then obtain the modified score

$$\epsilon_k \longleftarrow \tilde{\epsilon}_k - \sqrt{1-\bar{\alpha}_k}\ g_k$$

where $g_k = -\beta_k \sum_{i=1}^{2} w_i \nabla_{\mathbf{q}_k} \Big[1 - \mathcal{M}_i(y|\eta_i, \hat{\mathbf{q}}_0^k, \Phi)\Big]^2$. We notice that injecting noise to g_k, as in stochastic DDIM, can slightly improve the performance. We calculate the final $\mathbf{q}_{k-1}$ using the refined ϵ_k in (3). A visual clarification of the forward and reverse diffusion is shown in Fig. 3.

IV. Results: Simulation

Based on the environment setup as discussed in Section II, we create datasets, train diffusion and feasibility score models and evaluate them in simulation for proper placement conditions.

A. Performance Evaluation: Diffusion with Guidance

The trained classifier-free conditional diffusion model and the score feasibility models are used to perform the reverse diffusion using the classifier-free guidance with and without feasibility score guidance. Experiments comparing the performance of both the methods are shown in Fig. 4 for a set of YCB Objects [2] and different scene-task scenarios where only 50 candidate poses are sampled and top 10 high-likelihood poses are selected. The comparison shows that while the classifier-free guidance is good enough to sample high-likelihood reorientation poses, the primary purpose of the feasibility score gradients is to reduce the variance in the pose generation and ensure high success probability. A numerical analysis of the overall success is shown and compared with the rejection sampling based baseline [19] in Table I.

TABLE I: Success evaluation of the proposed method as compared to the rejection sampling based baseline. The ReorientDiff algorithm was tested for more than 300 different scene task settings consisting of equal distribution of the selected objects and all the orientations. A task is considered a success if it is completed at-least once in 4 random seeds.

Method	**Success (%) Reorient**	**Success (%) Place**	**Success (%) Overall**
ReorientBot	97.9	95.1	93.2
ReorientDiff (w/o Guide)	97.4	86.3	85.8
ReorientDiff	**98.9**	**96.5**	**94.8**

The reorientation success percentage holds different relevance as compared to the baseline. The baseline does two step reverse rejection sampling where reorientation search is conducted over candidates which are feasible for placement, so there might be a scenario where there is no solution possible. For the case of ReorientDiff, the reorientation success measures the capability of the diffusion model to generalize to poses which ensure reorientability and scope for future placement.

Higher reorientation success and lower placement success is an indication that the model is short-sighted and is giving importance to a single step success metric. From Table I, we ensure high reorientability success along with better placement success, even without any candidate pose discretization. The overall success is based on the accurate placement of the object from the reoriented pose, and it represents the successful completion of a task. The metric is measured by calculating the difference between the desired and the pose after final placement.

V. Conclusion

Diffusion models are powerful generative models capable of modeling (conditional) distributions. In the proposed method, ReorientDiff exploits the capabilities of such models to predict reorientation poses conditioned on a compact scene-task representation embedding containing information about the target object and its placement location. Further, the samples are refined using learned feasibility-score models to reduce uncertainty and ensure success of the planned intermediate poses. Considering as little as 10 reorientation poses, we achieved an overall success rate of 96.5% across variety of objects. We consider incorporating more efficient sampling schemes and better generalization performance for unseen objects and placement goals as a potential future work.

References

[1] Anurag Ajay, Yilun Du, Abhi Gupta, Joshua Tenenbaum, Tommi Jaakkola, and Pulkit Agrawal. Is conditional generative modeling all you need for decision-making? *arXiv preprint arXiv:2211.15657*, 2022.

[2] Berk Calli, Arjun Singh, Aaron Walsman, Siddhartha Srinivasa, Pieter Abbeel, and Aaron M Dollar. The ycb object and model set: Towards common benchmarks for manipulation research. In *2015 international conference on advanced robotics (ICAR)*, pages 510–517. IEEE, 2015.

[3] Erwin Coumans and Yunfei Bai. Pybullet, a python module for physics simulation for games, robotics and machine learning. http://pybullet.org, 2016–2021.

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

[5] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[6] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[7] Michael Janner, Yilun Du, Joshua Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. In *International Conference on Machine Learning*, 2022.

[8] Weiyu Liu, Tucker Hermans, Sonia Chernova, and Chris Paxton. Structdiffusion: Object-centric diffusion for semantic rearrangement of novel objects. *arXiv preprint arXiv:2211.04604*, 2022.

[9] Weiyu Liu, Chris Paxton, Tucker Hermans, and Dieter Fox. Structformer: Learning spatial structure for language-guided semantic rearrangement of novel objects. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 6322–6329. IEEE, 2022.

[10] Arsalan Mousavian, Clemens Eppner, and Dieter Fox. 6-dof graspnet: Variational grasp generation for object manipulation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2901–2910, 2019.

[11] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[12] Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Cliport: What and where pathways for robotic manipulation. In *Conference on Robot Learning*, pages 894–906. PMLR, 2022.

[13] Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Perceiver-actor: A multi-task transformer for robotic manipulation. *arXiv preprint arXiv:2209.05451*, 2022.

[14] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[15] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[16] Ioan A. Şucan, Mark Moll, and Lydia E. Kavraki. The Open Motion Planning Library. *IEEE Robotics & Automation Magazine*, 19(4):72–82, December 2012. doi: 10.1109/MRA.2012.2205651. https://ompl.kavrakilab.org.

[17] Bingjie Tang and Gaurav S. Sukhatme. Selective object rearrangement in clutter. In *6th Annual Conference on Robot Learning*, 2022. URL https://openreview.net/forum?id=2cEjfernc5P.

[18] Julen Urain, Niklas Funk, Georgia Chalvatzaki, and Jan Peters. Se (3)-diffusionfields: Learning cost functions for joint grasp and motion optimization through diffusion. *arXiv preprint arXiv:2209.03855*, 2022.

[19] Kentaro Wada, Stephen James, and Andrew J Davison. Reorientbot: Learning object reorientation for specific-posed placement. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 8252–8258. IEEE, 2022.

[20] Andy Zeng, Pete Florence, Jonathan Tompson, Stefan Welker, Jonathan Chien, Maria Attarian, Travis Armstrong, Ivan Krasin, Dan Duong, Vikas Sindhwani, et al. Transporter networks: Rearranging the visual world for robotic manipulation. In *Conference on Robot Learning*, pages 726–747. PMLR, 2021.

[21] Qinsheng Zhang and Yongxin Chen. Fast sampling of diffusion models with exponential integrator. *arXiv preprint arXiv:2204.13902*, 2022.

VI. Preliminary: Diffusion Models

Consider samples x_0 from an unknown data distribution $q(x_0)$; diffusion models [6] learn to estimate the distribution by a parameterized model $p_\theta(x_0)$ using the given samples. The procedure is completed in two steps: the forward and the reverse diffusion processes. The former continuously injects Gaussian noise in x_0 to create a Markov chain with latents $x_{1:K}$ following transitions:

$$q(x_{1:K}|x_0) = \prod_{k=1}^{K} q(x_k|x_{k-1}), \tag{6}$$

where $q(x_k|x_{k-1}) = \mathcal{N}(x_k; \sqrt{1-\beta_k}x_{k-1}, \beta_k\mathbf{I})$ is the per-step noise injection following variance schedule $\beta_1, \ldots, \beta_K$. This leads to the distribution $q(x_k|x_0) = \mathcal{N}(x_k; \sqrt{\bar\alpha_k}x_0, (1-\bar\alpha_k)\ \mathbf{I})$ following notations introduced in [14] as $\alpha_k = 1-\beta_k$ and $\bar\alpha_k = \prod_{i=1}^{k}\alpha_i$. Note that $\bar\alpha_K \approx 0$ and thus $x_K \sim \mathcal{N}(0, \mathbf{I})$. The reverse diffusion learns to denoise the data starting from x_K and following $p_\theta(x_{k-1}|x_k) = \mathcal{N}(x_{k-1}; \mu_\theta(x_k, k), \beta_k\mathbf{I})$ where

$$\mu_\theta(x_k, k) = \frac{1}{\sqrt{\alpha_k}}\Big(x_k - \frac{\beta_k}{\sqrt{1-\bar\alpha_k}}\epsilon_\theta(x_k, k)\Big). \tag{7}$$

The parameterized model $\epsilon_\theta(x_k, k)$ is called the score-function, and it is trained to predict the perturbations and the noising schedule by the score-matching objective [15]

$$\arg\min_\theta \mathbb{E}_{x_0\sim q, \epsilon\sim\mathcal{N}(0,\mathbf{I})}\Big[\|\epsilon - \epsilon_\theta(\sqrt{\bar\alpha_k}x_0 + \sqrt{1-\bar\alpha_k}\epsilon)\|^2\Big] \tag{8}$$

In particular, such a score function represents the gradients of the learned probability distribution as

$$\nabla_{x_k}\log p_\theta(x_k) = -\frac{1}{\sqrt{1-\bar\alpha_k}}\epsilon_\theta(x_k, k). \tag{9}$$

VII. Additional Result and Discussion

A. Dataset Generation and Training

We use PyBullet [3] and an OMPL [16] based motion planner to solve for collision-free path between current pose and a candidate reorientation pose and from the reorientation pose to the ground-truth placement pose for diverse set of YCB-objects and target locations. We sampled approximately 40000 candidate poses following Wada *et al.* [19]. The goal properties were converted into modular language instructions, and the success of pick and place for both the steps was recorded. The scene and task properties were used to construct the joint visual-language embedding space, which was further used to train the feasibility score models using binary success labels. Eventually, we train a conditional diffusion model using only the successful reorientation poses. Such a diffusion model is reusable for diverse set of grasp poses based on the feasibility score models.

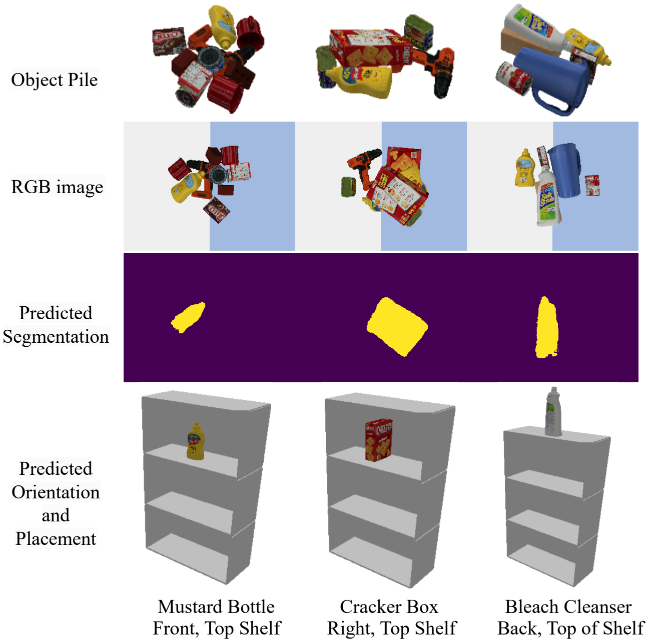


Fig. 5: **Visual Analysis of Scene-Task Network Performance** The scene-task network maps the visual (row 2) image of the pile (row 1) and language (bottom row) inputs to a feature space which is used to predict the placement location (row 4) and target object segmentation (row 3).

B. Performance Evaluation: Scene-Task Representation

To evaluate the quality of the scene-task embedding network, we analyze the accuracy of the object selection and placement pose prediction along with the error in the predicted segmentation. We show a visual analysis in Fig. 5 where the output segmentation and the predicted placement pose in the shelf are shown for three scenes and tasks. For accurate shelf-level estimation, we round each object's predicted height to the nearest shelf-level height, and a similar post-processing is conducted for the object orientation. To add complexity, although we consider only four orientations: front, back, left and right, we discretize the possible orientations into 8 possible options and round the predicted orientation to the nearest option.

Numerically, the object selection network was 100% accurate, and the number of pixels wrongly classified was about 1% of the complete image on average over 100 random samples. The average error in predicting the height of the target placement after post-processing is around 8 mm, and the mean error in the yaw angle of the predicted pose is 0.3 rad.

C. Performance Evaluation: K-Step Reverse Diffusion

Sampling from a trained diffusion models is flexible and can be achieved using different levels of discretization between $x_K \sim \mathcal{N}(0, \mathbf{I})$ to meaningful reorientation poses. We perform the complete analysis for multiple values of the number of reverse denoising steps K as shown in Table II. It is evident that minimizing resolution degrades overall performance, but even with much fewer resolutions, ReorientDiff can reach a descent performance.

TABLE II: Success evaluation with different levels of discretization while sampling using ReorientDiff.

ReorientDiff K	**Success (%)** **Reorient**	**Success (%)** **Place**	**Success (%)** **Overall**
$K = 256$	**98.9**	**96.5**	**94.8**
$K = 100$	99.3	92.4	91.5
$K = 50$	97.5	91.1	88.6

Following our analysis on performance, we explored the time taken for overall planning of a successful reorientation pose from a given scene and corresponding task information. We provide the recorded timings for all of our ablations as well as the baseline in Table III.

TABLE III: Computational analysis of the planning time for finding a suitable reorientation pose for the proposed method, ReorientDiff, along with the baseline and all conducted ablations.

Method	**Planning** **Time (sec)**
ReorientBot	2.5
ReorientDiff (w/o Guide)	2.7
ReorientDiff @ $K = 256$	**5.3**
ReorientDiff @ $K = 100$	2.5
ReorientDiff @ $K = 50$	1.5

Our findings show that ReorientDiff is computationally heavy due to gradient calculations for reverse denoising steps. Without using the guidance from the feasibility-score models, classifier-free guidance requires similar time as the baseline, ReorientBot. However, as we decrease the discretization resolution, the planning time decreases significantly with some trade-off in performance, as shown in Table II. We believe that using higher-order solvers such as one proposed in [21], similar level of performance as ReorientDiff (w/ $K = 256$) can be achieved at the computation cost of $K = 50$. However, such an analysis is out of scope of the proposed methodology. Hence, from all of our visual and empirical analysis, ReorientDiff successfully proves that formulating the problem of reorientation as learning a conditional distribution is an effective way to move towards more generalizable object manipulation.