# DALL-E-Bot:
# Introducing Web-Scale Diffusion Models to Robotics

**Ivan Kapelyukh**$^{*1,2}$, **Vitalis Vosylius**$^{*1}$, **Edward Johns**$^{1}$
$^{*}$ Joint first authorship
$^{1}$The Robot Learning Lab, $^{2}$The Dyson Robotics Lab
Imperial College London
United Kingdom
{ivan.kapelyukh17, vitalis.vosylius19, e.johns}@imperial.ac.uk

**Abstract:** We introduce the first work to explore web-scale diffusion models for robotics. DALL-E-Bot enables a robot to rearrange objects in a scene, by first inferring a text description of those objects, then generating an image representing a natural, human-like arrangement of those objects, and finally physically arranging the objects according to that image. The significance is that we achieve this zero-shot using DALL-E, without needing any further data collection or training. Encouraging real-world results with human studies show that this is a promising direction for the future of web-scale robot learning. We also propose a list of recommendations to the text-to-image community, to align further developments of these models with applications to robotics. Videos are available on our webpage at: https://www.robot-learning.uk/dall-e-bot

**Keywords:** Diffusion Models, Image Generation, Object Rearrangement

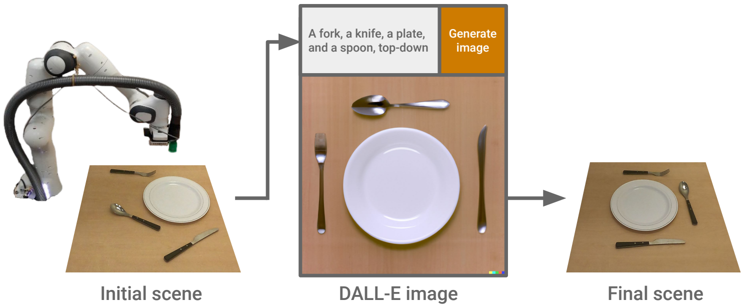

Figure 1: The robot prompts DALL-E with the list of objects it detects, which generates an image with a human-like arrangement of those objects. The robot then creates that arrangement in reality.

## 1 Introduction

Diffusion models such as DALL-E [1] have recently shown an astonishing ability to generate high-quality images from text prompts, through unsupervised training on millions of captioned images from the web [2, 3, 4]. Previous breakthroughs in web-scale foundation models have been applied successfully to robotics [5, 6, 7]. In this work, we explore the following question: **How can web-scale image diffusion models such as DALL-E be used for robotics?**

Since these models can generate realistic images of everyday scenes such as kitchens and offices, our insight is that they are proficient at imagining arrangements of everyday objects which are *human-like*: semantically correct, aesthetically pleasing, physically plausible, and convenient to use. Therefore, we consider that they could be used to generate goal images for generic object rearrangement

CoRL 2022 Workshop on Pre-training Robot Learning, Auckland, New Zealand.

tasks [8], such as setting a table, loading a dishwasher, tidying a room, stacking a shelf, and assembling furniture. Most prior methods for predicting the goal state (i.e. a set of goal poses for each object) require manually collecting a dataset of examples for how a scene should be arranged [9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19]. Our proposed framework predicts how to arrange a given scene without requiring this data collection, which restricts most existing methods to a specific set of objects and scenes. Further analysis of prior work can be found in Appendix A.

In this paper, we propose DALL-E-Bot, the first method to use web-scale image diffusion models for robotics. We design a framework which takes an image of the initial, unorganised scene, uses DALL-E to imagine a human-like goal image for that scene, and creates the corresponding object arrangement with a real robot (Fig. 1). Experiments show that this can be applied to several everyday rearrangement tasks to create arrangements which are satisfactory to humans. Additionally, we find that DALL-E's inpainting feature can precisely predict the poses of missing objects in a scene, conditioned on the pre-placed objects. Furthermore, we present a discussion of the method's limitations in Appendix K, and in Appendix L we propose ideas for future web-scale diffusion models to maximise their usefulness for robotics.

Using web-scale image diffusion models for predicting goal states in this way has several strengths. First, this is a *zero-shot* transfer of the DALL-E model to the object rearrangement task, because it uses the publicly available DALL-E without any additional data collection or training. Second, this is an *open-set* method: it is not restricted to a specific set of objects, because of the web-scale training of DALL-E. Third, this pipeline is *autonomous*: no human effort is required from the user, because there is no need for a human-created goal image or language guidance.

## 2 Method

We address the problem of predicting the goal state of a rearrangement task, i.e. a goal pose for each object, such that the objects are arranged in a natural and human-like way. The method must predict this goal state from a single RGB image $I_I$ of the initial scene. We achieve this through a modular approach shown in Fig. 2. At the heart of our method is a web-scale image diffusion model DALL-E 2 [1], which generates high-quality variations of images $I_G$ with human-like object arrangements using a language description of the scene $\ell$ extracted from the initial observation.

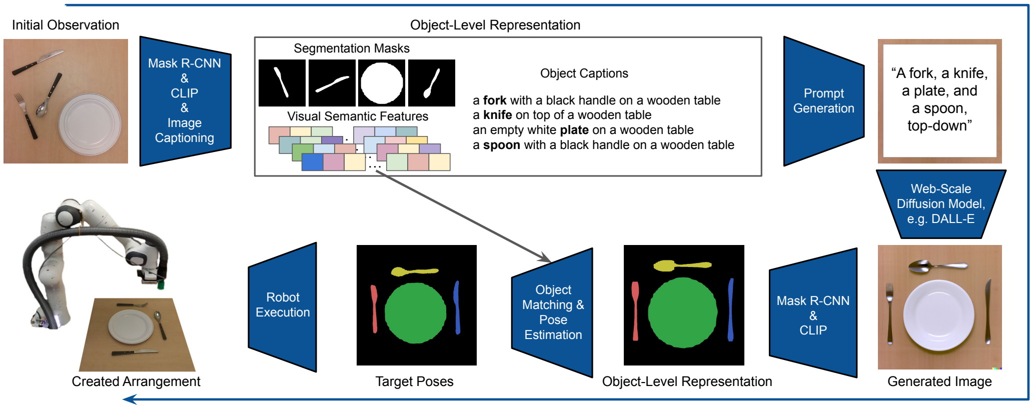
Initial Observation
Mask R-CNN & CLIP & Image Captioning
Object-Level Representation
Segmentation Masks
Visual Semantic Features
Object Captions
a fork with a black handle on a wooden table
a knife on top of a wooden table
an empty white plate on a wooden table
a spoon with a black handle on a wooden table
Prompt Generation
"A fork, a knife, a plate, and a spoon, top-down"
Web-Scale Diffusion Model, e.g. DALL-E
Generated Image
Mask R-CNN & CLIP
Object-Level Representation
Object Matching & Pose Estimation
Target Poses
Robot Execution
Created Arrangement

Figure 2: An overview of our method's pipeline.

First, we need to convert an initial RGB observation into a more relevant object-level representation to reason about the objects in the scene and their arrangement. We do so by constructing a representation that consists of text captions of crops of individual objects $c_i$ in the scene together with their segmentation masks $M_i$ and visual-semantic feature vector $v_i$ acquired using the CLIP model [20].

We use text captions $c_i$ to automatically construct a text prompt containing a list of the objects in the scene. We also append the term ”top-down” so that DALL-E generates images from the same perspective as the initial image captured by a camera mounted on a robot's wrist pointing downwards better. In addition, we generate an image mask $I_M$ that prevents DALL-E from altering the pixels corresponding to the contours of stationary objects (i.e. an object that the robot is not allowed to move) and tabletop edges to avoid objects being generated on the edge of the image.

We generate several images with the goal arrangement by sampling a conditional distribution $p_\theta(I_G|\ell, I_M)$ represented by a web-scale text-to-image diffusion model DALL-E 2 [1]. We convert generated images into object-level representations and filter out the ones that do not contain the same number of objects as the initial scene. From the remaining images, we select the one that minimises the cost of the linear sum assignment problem (Hungarian matching) between the visual-semantic feature vectors in the initial and generated images.

Using Iterative Closest Point (ICP) [21], we then register corresponding segmentation masks to obtain transformations that need to be applied to the objects to achieve the goal arrangement. To account for possible size differences for the same object in initial and generated images, we move objects closer together or further apart, but do not allow them to collide. Finally, we convert these transformations from image to Cartesian space using a depth camera observation and deploy a real Franka Emika Panda robot equipped with a suction gripper to arrange the objects. More detailed explanations of each component in our method can be found in Appendices B-E.

# 3 Experiments

## 3.1 Zero-Shot Autonomous Rearrangement

In our experiments, we evaluate the ability of our method to create human-like arrangements using both subjective (Section 3.1) and objective (Section 3.2) metrics. First, we explore the following question: **can DALL-E-Bot arrange a set of objects in a human-preferred way?**

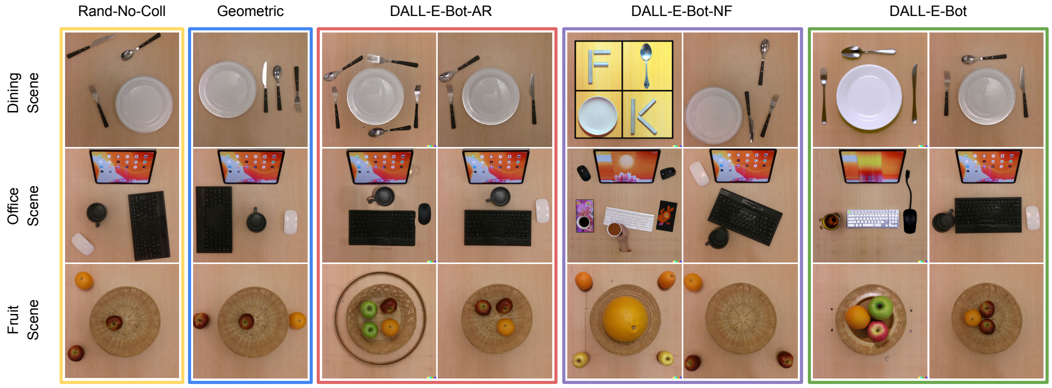

Figure 3: Examples of scenes rearranged by the robot using different methods. Columns for the methods that use DALL-E include the generated image (left) and the final arrangement (right). For DALL-E-Bot-AR, images are from the last step.

We evaluate on 3 everyday tabletop rearrangement tasks: **dining**, **office**, and **fruit** (Fig. 3). The robot should arrange the objects in a human-like way while considering the poses of fixed objects (the iPad and the fruit basket). Setup details are in Appendix G. Since DALL-E-Bot is the first method to predict precise goal states for rearrangement zero-shot, we design baselines which are also zero-shot for a fair comparison. We use heuristic baselines and variants of DALL-E-Bot, detailed in Appendix H.

| Method | Dining Scene | Office Scene | Fruit Scene | Mean |
|---|---|---|---|---|
| Rand-No-Coll | 2.03±1.34 | 3.56±2.01 | 2.94±2.01 | 2.84 |
| Geometric | 4.08±2.27 | 3.36±2.01 | 3.13±1.82 | 3.52 |
| DALL-E-Bot-NF | 3.87±2.78 | 6.54±2.34 | 7.45±3.19 | 5.95 |
| DALL-E-Bot-AR | 4.88±2.61 | 7.37±2.05 | 9.59±0.90 | 7.28 |
| DALL-E-Bot | **8.01**±2.03 | **7.56**±2.02 | **9.81**±0.52 | **8.46** |

Table 1: User ratings for the arrangements made by each method. Each figure represents the mean and standard deviation across all users and scene initialisations.

We evaluated by showing human users images of the final scene created by the robot. Users were asked: *"If the robot made this arrangement for you at home, how happy would you be?"*, with ratings on a Likert Scale from 1 (very unhappy) to 10 (very happy). We recruited 40 users representing 18 nationalities, both male and female, with ages ranging from 22 to 71. Each rated the results of 5 methods on 5 random initialisations of 3 scenes, for a total of 3000 ratings. Initialisations were roughly matched for all the methods and all users were shown the same images. Results are in Table 1. DALL-E-Bot beats the heuristic baselines, showing that **people value semantic correctness over simple geometric alignment**. DALL-E-Bot also consistently beats its variants in all of the evaluation scenes, justifying our design decisions. For a detailed analysis, please see Appendix J.

### 3.2 Placing Missing Objects with Inpainting

In the next experiment, we use objective metrics to answer the question: **can DALL-E-Bot precisely complete an arrangement which was partially made by a human?** For this, we ask DALL-E-Bot to find a suitable pose for an object that has been masked out from a user-made scene. We use the dining scene because it has the most rigid structure for semantic correctness and thus is most suitable for quantitative, objective evaluation. To create these scenes initially, we recruited ten participants (both left and right-handed) and asked them the following: "*Imagine you are sitting down here for dinner. Can you please arrange these objects so that you are happy with the arrangement?*". As there can be multiple suitable poses for any single object in the scene, we asked the users to provide any alternative poses of each object individually that they would still be happy with while keeping other objects fixed. We show example arrangements in Appendix I.

|  | Fork | Plate | Spoon | Knife |
|---|---|---|---|---|
| Method | cm / deg | cm / deg | cm / deg | cm / deg |
| Rand-No-Coll | 25.85 / 70.32 | 10.78 / - | 27.47 / 42.56 | 23.51 / 99.32 |
| Geometric | 15.59 / 40.57 | 2.29 / - | 23.83 / 86.11 | 11.58 / **1.47** |
| DALL-E-Bot | **4.95 / 1.26** | **1.28** / - | **2.13 / 2.72** | **2.1** / 3.27 |

Table 2: Position and orientation errors between predicted and user-made object poses. Median is presented across all users.

We start with the image of the arrangement made by a user, and mask out everything except the fixed objects. The method must then predict the pose of the missing object. DALL-E-Bot does this by inpainting the missing object somewhere in the image. For a given user, the predicted pose for the missing object is compared against the actual pose in their arrangement. This is done by aligning two segmentation masks of the missing object, one from the actual scene and one at a predicted pose. Since this is for two poses of exactly the same object instance, we find the alignment is highly accurate and can be used to estimate the error between the actual and predicted pose. From this transformation, we take the orientation and distance errors projected into the workspace as our metrics. This is repeated for every object as the missing object, and across all the users. We compare our method to two heuristic baselines (see Appendix H). In Table 2, we report the medians of translation and rotation errors to the closest placement of each object from the ones each separate user provided as being acceptable. DALL-E-Bot outperforms the heuristic baselines, and is able to accurately place the missing objects for different users. This implies that it is conditioning on the placement of the other objects in the scene using inpainting, and that the human and robot can create an arrangement collaboratively.

### 3.3 Conclusions

We have introduced the first method to use web-scale diffusion models for robotics. DALL-E-Bot enables zero-shot object rearrangement in everyday scenes, using DALL-E as an "imagination engine" for goal states. We believe that this is an exciting direction for the future of robot learning, as diffusion models continue to impress and inspire complementary research communities.

**Acknowledgments**

Research presented in this paper has been supported by Dyson Technology Ltd, and the Royal Academy of Engineering under the Research Fellowship Scheme. The authors would like to thank Andrew Davison, Ignacio Alzugaray, Kamil Dreczkowski, Kirill Mazur, Eric Dexheimer, and Tristan Laidlow for helpful discussions. The authors would also like to thank Kentaro Wada for developing some of the robot control infrastructure which was used in the experiments.

## References

[1] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen. Hierarchical text-conditional image generation with CLIP latents. *arXiv*, 2022.

[2] A. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew, I. Sutskever, and M. Chen. GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv*, 2021.

[3] 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, T. Salimans, J. Ho, D. J. Fleet, and M. Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv*, 2022.

[4] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. *arXiv*, 2021.

[5] M. Shridhar, L. Manuelli, and D. Fox. CLIPort: What and where pathways for robotic manipulation. In *Proceedings of the 5th Conference on Robot Learning (CoRL)*, 2021.

[6] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, C. Fu, K. Gopalakrishnan, K. Hausman, A. Herzog, D. Ho, J. Hsu, J. Ibarz, B. Ichter, A. Irpan, E. Jang, R. J. Ruano, K. Jeffrey, S. Jesmonth, N. J. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, K.-H. Lee, S. Levine, Y. Lu, L. Luu, C. Parada, P. Pastor, J. Quiambao, K. Rao, J. Rettinghouse, D. Reyes, P. Sermanet, N. Sievers, C. Tan, A. Toshev, V. Vanhoucke, F. Xia, T. Xiao, P. Xu, S. Xu, M. Yan, and A. Zeng. Do as I can, not as I say: Grounding language in robotic affordances. *arXiv*, 2022.

[7] W. Huang, F. Xia, T. Xiao, H. Chan, J. Liang, P. Florence, A. Zeng, J. Tompson, I. Mordatch, Y. Chebotar, P. Sermanet, N. Brown, T. Jackson, L. Luu, S. Levine, K. Hausman, and B. Ichter. Inner monologue: Embodied reasoning through planning with language models. *arXiv*, 2022.

[8] D. Batra, A. X. Chang, S. Chernova, A. J. Davison, J. Deng, V. Koltun, S. Levine, J. Malik, I. Mordatch, R. Mottaghi, M. Savva, and H. Su. Rearrangement: A challenge for embodied AI. *arXiv*, 2020.

[9] Y. Jiang, M. Lim, and A. Saxena. Learning object arrangements in 3D scenes using human context. *Proceedings of the 29th International Conference on Machine Learning, ICML 2012*, 2012.

[10] I. Kapelyukh and E. Johns. My house, my rules: Learning tidying preferences with graph neural networks. In *Conference on Robot Learning (CoRL)*, 2021.

[11] Y. Lin, A. S. Wang, E. Undersander, and A. Rai. Efficient and interpretable robot manipulation with graph neural networks. *IEEE Robotics and Automation Letters*, 7:2740–2747, 2022.

[12] V. Jain, Y. Lin, E. Undersander, Y. Bisk, and A. Rai. Transformers are adaptable task planners. In *6th Annual Conference on Robot Learning*, 2022.

[13] W. Liu, C. Paxton, T. Hermans, and D. Fox. StructFormer: Learning spatial structure for language-guided semantic rearrangement of novel objects. *2022 International Conference on Robotics and Automation (ICRA)*, 2022.

[14] A. Zeng, P. Florence, J. Tompson, S. Welker, J. Chien, M. Attarian, T. Armstrong, I. Krasin, D. Duong, V. Sindhwani, and J. Lee. Transporter networks: Rearranging the visual world for robotic manipulation. *Conference on Robot Learning (CoRL)*, 2020.

[15] G. Sarch, Z. Fang, A. W. Harley, P. Schydlo, M. J. Tarr, S. Gupta, and K. Fragkiadaki. TIDEE: Tidying up novel rooms using visuo-semantic commonsense priors. In *European Conference on Computer Vision*, 2022.

[16] M. Kang, Y. Kwon, and S.-E. Yoon. Automated task planning using object arrangement optimization. In *2018 15th International Conference on Ubiquitous Robots (UR)*, pages 334–341, 2018.

[17] A. Taniguchi, S. Isobe, L. E. Hafi, Y. Hagiwara, and T. Taniguchi. Autonomous planning based on spatial concepts to tidy up home environments with service robots. *Advanced Robotics*, 35 (8):471–489, 2021.

[18] M. J. Schuster, D. Jain, M. Tenorth, and M. Beetz. Learning organizational principles in human environments. In *2012 IEEE International Conference on Robotics and Automation*, pages 3867–3874, 2012.

[19] N. Abdo, C. Stachniss, L. Spinello, and W. Burgard. Robot, organize my shelves! Tidying up objects by predicting user preferences. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[20] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, Proceedings of Machine Learning Research, 2021.

[21] P. Besl and H. McKay. A method for registration of 3-D shapes. ieee trans pattern anal mach intell. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 1992.

[22] Y. Kant, A. Ramachandran, S. Yenamandra, I. Gilitschenski, D. Batra, A. Szot, and H. Agrawal. Housekeep: Tidying virtual households using commonsense reasoning. *arXiv*, 2022.

[23] C. Finn and S. Levine. Deep visual foresight for planning robot motion. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017.

[24] M. Babaeizadeh, M. T. Saffar, S. Nair, S. Levine, C. Finn, and D. Erhan. FitVid: Overfitting in pixel-level video prediction. *arXiv*, 2020.

[25] M. Wu, F. Zhong, Y. Xia, and H. Dong. TarGF: Learning target gradient field for object rearrangement. *arXiv*, 2022.

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

[27] A. Goyal, A. Mousavian, C. Paxton, Y.-W. Chao, B. Okorn, J. Deng, and D. Fox. IFOR: Iterative flow minimization for robotic object rearrangement. *arXiv*, 2022.

[28] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the 32nd International Conference on Machine Learning*, 2015.

[29] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, 2020.

[30] A. Nichol and P. Dhariwal. Improved denoising diffusion probabilistic models, 2021.

[31] P. Dhariwal and A. Nichol. Diffusion models beat GANs on image synthesis. In *Advances in Neural Information Processing Systems*, 2021.

[32] M. Janner, Y. Du, J. Tenenbaum, and S. Levine. Planning with diffusion for flexible behavior synthesis. In *International Conference on Machine Learning*, 2022.

[33] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro. DiffWave: A versatile diffusion model for audio synthesis. *ArXiv*, 2021.

[34] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. In *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017.

[35] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019.

[36] A. Gupta, P. Dollar, and R. Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[37] P. Wang, A. Yang, R. Men, J. Lin, S. Bai, Z. Li, J. Ma, C. Zhou, J. Zhou, and H. Yang. OFA: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *Proceedings of the 39th International Conference on Machine Learning*, 2022.

[38] A. Akbik, D. Blythe, and R. Vollgraf. Contextual string embeddings for sequence labeling. In *COLING 2018, 27th International Conference on Computational Linguistics*, 2018.

[39] A. Akbik, T. Bergmann, D. Blythe, K. Rasul, S. Schweter, and R. Vollgraf. FLAIR: An easy-to-use framework for state-of-the-art NLP. In *NAACL 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, 2019.

[40] W. Goodwin, S. Vaze, I. Havoutis, and I. Posner. Semantically grounded object matching for robust robotic scene rearrangement. In *2022 International Conference on Robotics and Automation (ICRA)*, 2022.

[41] T. Ridnik, E. Ben-Baruch, A. Noy, and L. Zelnik-Manor. ImageNet-21K pretraining for the masses. *arXiv*, 2021.

[42] T. Ridnik, H. Lawen, A. Noy, E. Ben, B. G. Sharir, and I. Friedman. TResNet: High performance gpu-dedicated architecture. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2021.

[43] J. J. Kuffner and S. M. LaValle. RRT-connect: An efficient approach to single-query path planning. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, volume 2, pages 995–1001. IEEE, 2000.

[44] E. Johns, S. Leutenegger, and A. J. Davison. Deep learning a grasp function for grasping under gripper pose uncertainty. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016.

[45] A. Mousavian, C. Eppner, and D. Fox. 6-DOF GraspNet: Variational grasp generation for object manipulation. In *International Conference on Computer Vision (ICCV)*, 2019.

[46] V. Vosylius and E. Johns. Where to start? Transferring simple skills to complex environments. In *6th Annual Conference on Robot Learning*, 2022.

[47] E. Johns. Coarse-to-fine imitation learning: Robot manipulation from a single demonstration. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

[48] K. Mazur, E. Sucar, and A. J. Davison. Feature-realistic neural fusion for real-time, open set scene understanding. *arXiv*, 2022.

[49] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa. Large language models are zero-shot reasoners. *ArXiv*, 2022.

[50] C. Conwell and T. Ullman. Testing relational understanding in text-guided image generation. *arXiv*, 2022.

[51] R. Gal, Y. Alaluf, Y. Atzmon, O. Patashnik, A. H. Bermano, G. Chechik, and D. Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv*, 2022.

[52] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman. DreamBooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv*, 2022.

[53] W. Hong, M. Ding, W. Zheng, X. Liu, and J. Tang. CogVideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv*, 2022.

[54] U. Singer, A. Polyak, T. Hayes, X. Yin, J. An, S. Zhang, Q. Hu, H. Yang, O. Ashual, O. Gafni, D. Parikh, S. Gupta, and Y. Taigman. Make-A-Video: Text-to-video generation without text-video data. *arXiv*, 2022.

[55] J. Ho, W. Chan, C. Saharia, J. Whang, R. Gao, A. Gritsenko, D. P. Kingma, B. Poole, M. Norouzi, D. J. Fleet, and T. Salimans. Imagen video: High definition video generation with diffusion models. *arXiv*, 2022.

[56] B. Poole, A. Jain, J. T. Barron, and B. Mildenhall. DreamFusion: Text-to-3D using 2D diffusion. *arXiv*, 2022.

# A Related Work

## A.1 Predicting Goal Arrangements

Here we highlight prior approaches to predicting goal poses for rearrangement tasks. Some methods view the prediction of goal poses as a classification problem, by choosing from a set of discrete options for an object's placement. For house-scale rearrangement, a pre-trained language model can be used to predict goal receptacles such as tables [22], and out-of-place objects can be detected automatically [15]. At a room level, the correct drawer or shelf can be classified [18], taking preferences into account [19]. Lower-level prediction from a dense set of goal poses can be achieved with a graph neural network [11] or a preference-aware transformer [12]. Our framework uses high-resolution images of how objects should be placed, thus not requiring a set of discrete options to be pre-defined, and predicting more precise poses than is possible with language. Prior robotics work has trained generative models for visual control [23, 24], but our work shows that web-scale models such as DALL-E can be used zero-shot, even for multi-stage rearrangement tasks.

Methods for predicting continuous object poses typically use a dataset of example arrangements. They can learn spatial preferences with a graph VAE [10], or model gradient fields [25]. For language-conditioned rearrangement, an autoregressive transformer [13] can be used, or a diffusion model over poses can be combined with learned discriminators [26]. Other methods use full demonstrations [5, 14], or leverage priors such as human pose context [9]. However, unlike these works, our proposed framework does not require collecting and training on a dataset of rearrangement examples, which often restricts these methods to a specific set of objects and scenes. Instead, exploiting existing web-scale image diffusion models enables zero-shot rearrangement. When the goal image is given, rearrangement is possible even with unknown objects [27]. Our method does not require a user-provided goal image, and is thus an autonomous system.

## A.2 Web-Scale Diffusion Models

Generating images with web-scale diffusion models such as DALL-E is at the heart of our method. Diffusion models [28] are trained to reverse a single step of added noise to a data sample. By starting from random noise and iteratively running many of these small, learned denoising steps, this can generate a sample from the learned distribution of data. These models have been used to generate images [29, 30, 31], text-conditioned images [1, 2, 3, 4], robot trajectories [32], and audio waves [33]. We use DALL-E 2 [1] in this work, although our framework could be used with other text-to-image models.

# B Object-Level Representation

To reason about the poses of individual objects in the observed scene, we need to convert the initial RGB observation into a more functional, object-level representation. We use the Mask R-CNN model [34] from the Detectron2 library [35] to detect objects in an image and generate segmentation masks $M_i$. This model was pre-trained on the LVIS dataset [36], which has 1200 object classes, being more than sufficient for many rearrangement tasks. The Mask R-CNN model provides us with object bounding boxes, their segmentation masks and class labels. However, while bounding box and segmentation mask predictions are usually high-quality (regardless of the predicted class), and can be used for pose estimation (described in Section E), the assigned class labels are often incorrect due to the large number of classes in the training dataset.

As we are using text labels of objects in the scene (described in Section C) to construct a prompt for an image diffusion model, it is crucial for these labels to be accurate and descriptive. Instead of directly using predicted object class labels, we pass RGB crops of each object individually through the OFA image-to-text captioning model [37] and acquire a text description of the objects in the initial scene observation $c_i$. Generally, this approach allows us to more accurately predict object class labels and go beyond the objects in the training distribution and even obtain their visual characteristics such as colour, material and shape.

Finally, we also pass each object crop through a CLIP visual model [20], giving each object a 512-dimensional visual-semantic feature vector $v_i$. These features will be used later for matching objects between the initial scene image and the generated image. Thus we have converted an initial scene RGB observation $I_I$ into an object-level representation of the scene $(M_i, c_i, v_i)$, with a segmentation mask, a text caption, and a semantic feature vector for each object.

# C Goal Image Generation

Our method relies on the ability to generate images of natural and human-friendly arrangements given their language descriptions. To this end, we heavily utilise the recent advances in text-to-image generation using web-scale diffusion models. Specifically, we use the DALL-E 2 [1] model from OpenAI. It was trained on a vast number of image-caption pairs from the Web, and represents the conditional distribution $p_\theta(I_G|\ell, I_M)$. Here, $I_G$ is an image generated by the model, $\ell$ is a text prompt, and $I_M$ is an image mask that can be used to prevent the model from changing the values of certain pixels in the image. A large portion of distribution $p_\theta$ represents images with scenes arranged by humans in a friendly and usable way. Therefore, by sampling this distribution, we can generate images representing our desired scenes and create the object arrangements by matching the object poses in them. Additionally, the ability to condition this distribution on image mask $I_M$ allows us to tackle scenarios where not all objects in the scene need to or can be moved by the robot.

We first need to construct a text prompt $\ell$ describing the desired scene. To this end, we use object captions from our object-level representation. Although full captions, including visual characteristics, could be used to generate images with objects closely resembling the observed ones, in this work, we only use the nouns describing the object's class and leave including visual characteristics for future work. We extract the class of each object from the caption of its object crop, i.e. we extract "apple" from "a red apple on a wooden table". We do this by passing the object captions through the Part-of-Speech tagging model [38] from the Flair NLP library [39], which tags each word as a noun, a verb, etc. From this list of classes, we construct a prompt that makes minimal assumptions about the scene to allow DALL-E to arrange it in the most natural way. This work deals with tabletop scenes with initial observations captured by a camera mounted on a robot's wrist pointing downwards. Therefore, we added a "top-down" phrase to the prompt to better align the initial and generated images. We have also found that it reduces the frequency of generated images with unusual, artistic camera perspectives. An example prompt we use would be "A fork, a knife, a plate, and a spoon, top-down".

We use the ability to condition distribution $p_\theta$ on image masks in three ways. First, if there are objects in the scene that a robot is not allowed to move, we add their contours to $I_M$. This prevents DALL-E from generating these objects in different poses while still allowing for other objects to be placed on top or in them (e.g. a basket can not be moved, but other objects can be placed inside it). Secondly, we add a mask of the tabletop's edges in our scene to $I_M$ to **visually ground** the generated images. This prevents objects from being placed on the edge of the generated image and incentivises DALL-E to create objects of appropriate sizes. Finally, we subtract enlarged segmentation masks of all the movable objects from $I_M$ to avoid any shadows. The latter is essential, as if DALL-E sees any shadows of objects in their original poses, it will generate objects in the same poses to match the shadows, hindering the method's performance.

Using the prompt $\ell$ and the conditional mask $I_M$, we sample a batch of images from the conditional distribution $p_\theta(I_G|\ell, I_M)$, represented by the text-to-image model. We do so using an automated script and OpenAI's web API.

# D Image Selection & Object Matching

In the batch of generated images, not all will be desirable for the rearrangement task: some may have artefacts which make object detection difficult, others may contain the wrong number of objects, etc. We need to select the generated image $I_G$ which best matches the real-world initial image $I_I$.

For each generated image, we obtain segmentation masks and a CLIP semantic feature vector for each object using the same procedure as in Section B. We filter out generated images with the wrong number of objects, compared to the initial scene. Then, we match the objects in the generated image to the objects in the initial image. This is non-trivial since the generated objects are different instances to the real objects, with a very different appearance. Inspired by [40], a similarity score between any two objects (one from $I_I$, and one from $I_G$) is computed using the cosine similarity between their CLIP feature vectors. Since greedy matching is not guaranteed to yield optimal results in general, we use the Hungarian Matching algorithm to compute an assignment of each object in the live image to an object in the generated image, such that the total similarity score is maximised. Then we select the generated image $I_G$ which has the best overall score with the initial image $I_I$. This image contains the most similar set of objects to the real scene, and so that arrangement is most likely to transfer well to the real objects.

# E Object Pose Estimation

For each object in the initial image, we now know its segmentation mask in the initial image and the corresponding segmentation mask in the generated image. By aligning these masks, we can estimate a transformation from the initial pose (in the initial image) to the goal pose (in the generated image). We rescale each initial segmentation mask, such that the dimensions of its bounding box equal those in the generated image, and then use the Iterative Closest Point algorithm [21] to align the two masks, taking each pixel to be a point. This gives us a 3-DoF $(x, y, \theta)$ transformation $\mathcal{T}$ in pixel space between the initial and goal pose. We run ICP from many random initial poses, to handle local optima. For objects with nearly symmetric binary masks such as knives, aligning masks with ICP leads to multiple candidate solutions (for knives, they differ by 180 degrees). To select the correct solution (handle aligned with handle, blade aligned with blade), we pass the generated object image $o_G$ and the transformed real object image $\mathcal{T}(o_I)$ through a semantic feature map extractor $f_S$ (an ImageNet-trained ResNet [41, 42]). We select the ICP solution $\mathcal{T}$ which minimises the photometric loss between the semantic feature maps: $\mathcal{L}_S = (f_S(o_G) - f_S(\mathcal{T}(o_I)))^2$.

The scale of the objects in the generated image can be significantly different to the initial image, leading to the predicted arrangement resulting in collisions, or being unnaturally spaced out. Therefore, we adjust the poses of the objects in the scene based on the size difference of objects between the initial and generated images. We move all the objects closer to or further from the one with the minimum cumulative distance to all the other ones. Additionally, if collisions occur in the found arrangement, we move objects away from the central one until there are no more collisions.

Next, we use a wrist-mounted depth camera to project the pixel-space poses into 3D space on the tabletop, to obtain a transformation for each object which would move it from the initial real-world pose to the goal real-world pose. Finally, the robot executes these transformations by performing a sequence of pick-and-place operations using a suction gripper.

# F Robot Execution

Although the core of our contribution is predicting target poses for objects, we also construct a pick-and-place robot pipeline to evaluate our framework in the real world. For each object, the robot arm grasps the object in its initial pose, and moves it to its target pose, performing a rotation of $\theta$ in between, to achieve the target orientation. The robot's motion is calculated using Inverse Kinematics and interpolating Cartesian end-effector poses between a series of waypoints that move the robot to $pre-grasp$, $grasp$, $pre-place$, and $place$ poses. We define $pre-grasp$ and $pre-place$

poses as being $15$ centimetres higher than the $grasp$ and $place$ poses, respectively. If end-effector motion in a linear Cartesian path is not possible due to kinematic or collision constraints, we use the RRT-Connect [43] motion planner to find a collision-free path between the waypoints.

We sample many possible pick and place orientations for the gripper, and select one which satisfies kinematic and motion planning constraints. A grasping primitive is used for objects, to allow rearrangement with our suction gripper set-up. Most objects are grasped by their centre of mass, but if the eccentricity of their masks exceeds a threshold (i.e. they are elongated like cutlery), then they are grasped by the handle instead. The handle part of an object is determined from the principal axis of its mask, by choosing the object's "tail" using the skew of the mask. Grasping is not a core contribution of this paper, so for more difficult objects or cluttered environments the grasping primitive can be replaced with more complex methods [44, 45, 46]. If these transforms were executed naively, then an object being placed into its goal pose may collide with another object still in its initial pose. Therefore, we check for collisions before performing the pick-and-place actions, and move objects out of the way first if required to intermediate slots on the side of the table, outside the image. After the other objects have been placed, the robot also places the objects still in the intermediate slots into their target poses, thus completing the rearrangement and realising the generated image in the physical world.

## G Evaluation Setup

The **dining scene** involves four objects (a knife, a fork, a spoon, and a plate), and a robot should be able to arrange them so that a user would be happy seeing said arrangement when sitting down for a meal. The **office scene** includes a stationary object (a display) and three movable objects (a keyboard, a mouse and a mug). The arrangement of movable objects should be natural and useable with respect to the stationary object that a robot cannot move. Finally, the **fruit scene** contains two apples and an orange, as well as a stationary basket. This scene is challenging because it requires reasoning about the spatial relations between the fruits and the basket, and because the fruit in the generated images is often densely packed partially occluding the basket. The rearrangements are executed on a Franka Emika robot equipped with a compliant suction gripper. We record the outcome as an RGB image of a tabletop captured by RealSense D435i mounted on the wrist of the robot. Note that initialisations of the scene were roughly matched for all the methods being compared, and all users in the user study were shown the same images.

## H Baselines

### H.1 Zero-Shot Autonomous Rearrangement

Since DALL-E-Bot is the first method to predict arrangements zero-shot, we devised additional training-free methods as baselines, which can create arrangements that are natural to humans in our evaluation scenes. The Rand-No-Coll arrangement strategy arbitrarily places objects in the environment while ensuring they do not overlap. The Geometric baseline puts all the objects in a horizontal line such that they are not colliding, and the longer side of the object-oriented bounding box is aligned with the y-axis. In addition, we compare our method DALL-E-Bot to two different variants. DALL-E-Bot-NF (no filtering) does not filter generated images and always uses the first DALL-E generated image. If the image has fewer objects than the live scene, unmatched objects are placed randomly, ensuring there are no collisions. DALL-E-Bot-AR creates an arrangement in an auto-regressive way by moving one object at a time (from biggest to smallest) and treating it as a fixed object in the next iteration. The arrangement is created progressively around real objects. Therefore, it does not adjust the poses of the objects based on the size mismatch and does not reject generated images with a wrong number of objects. Examples of DALL-E generated images and achieved arrangements during autoregressive steps in our evaluation scenes can be seen in Fig. 4-6.

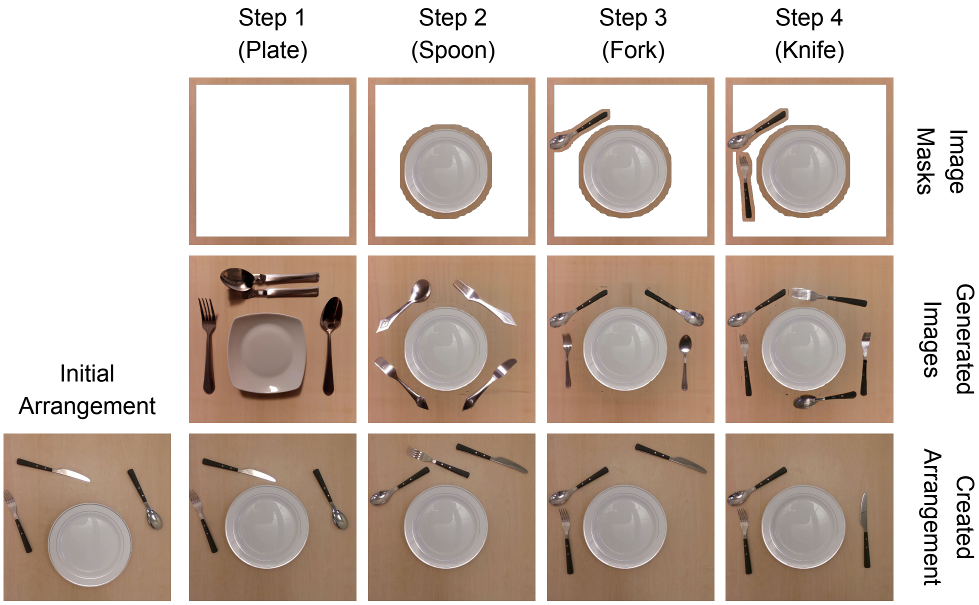

Figure 4: An example of DALL-E-Bot-AR execution in the Dinning Scene.

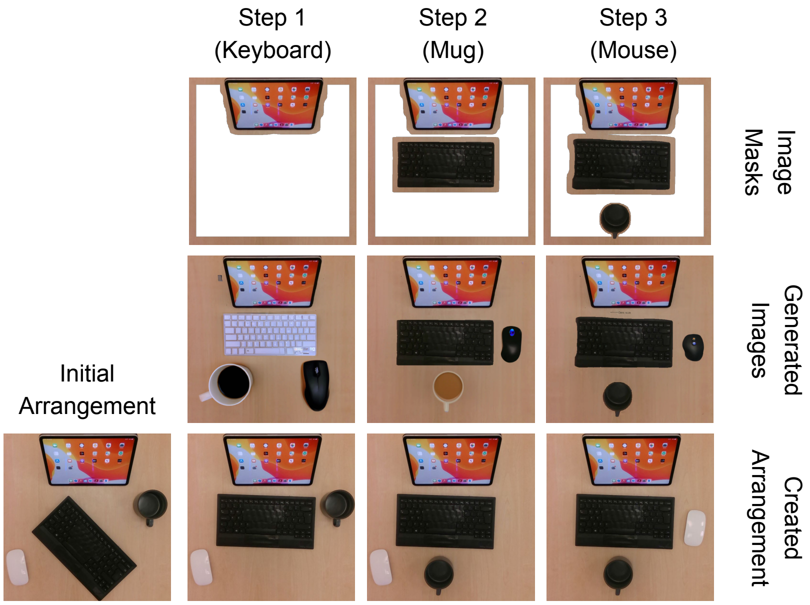

Figure 5: An example of DALL-E-Bot-AR execution in the Office Scene.

### H.2 Placing Missing Objects with Inpainting

Hand-designed baselines (Rand-No-Coll and Geometric) aim to place the missing object in a geometrically pleasing way based on the poses of other objects in the scene.

The Rand-No-Coll approach places the missing object arbitrarily in the workspace, ensuring it does not collide with the fixed objects. The Geometric baseline places the object on a line defined by centroids of segmentation maps of two fixed objects while also matching the alignment of the closest object.

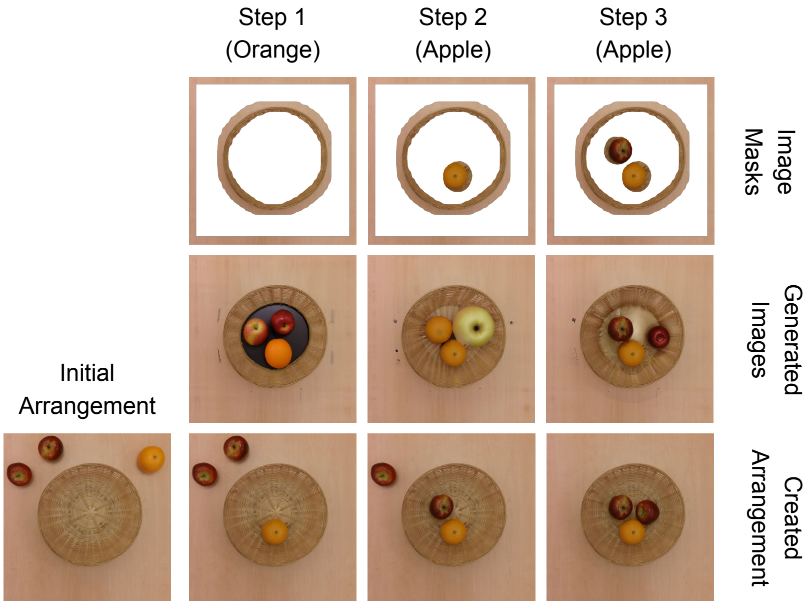

Figure 6: An example of DALL-E-Bot-AR execution in the Fruit Scene.

The distribution of acceptable poses is multimodal, which can cause significant errors if a method finds a mode not selected by the user. Therefore, we present the median across all users, which is less dominated by outliers than the mean, so it is a better representation of the aggregate performance.

# I User-Provided Arrangements for Inpainting

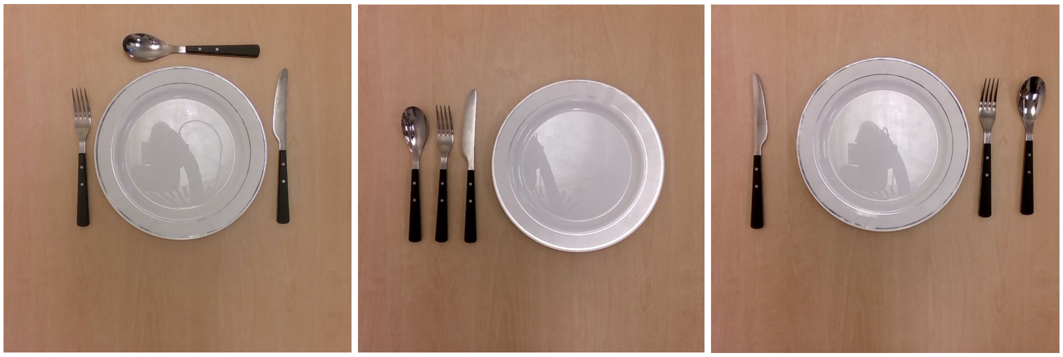

Figure 7: Example arrangements made by users for the inpainting experiment.

In the inpainting experiment, we ask users to create example arrangements so that methods can predict the poses of masked-out objects. In Fig. 7, we visualise several of the example arrangements provided by users. Even for a scene with as much semantic structure as a dining table, there is still significant variation in how users arrange this scene, due to their national cultural background or personal preferences. This shows that the methods benefit from conditioning on the placement of the pre-placed objects in order to place the missing object correctly. It also justifies our evaluation methodology for handling this multi-modal distribution, where we ask the users to provide several example placements for an object if they consider them all acceptable, and methods should predict any of these to achieve a low error.

# J Experimental Results Discussion

## J.1 Zero-Shot Autonomous Rearrangement

Looking at the user studies results presented in Table 1 in the main paper, we can see that DALL-E-Bot receives higher user scores, showing that it can create satisfactory arrangements even without task-specific training. Note that DALL-E has likely never seen these specific object instances before. DALL-E-Bot beats the heuristic baselines, showing that users do care about semantic correctness for arranging scenes beyond just geometric alignment, and justifying the use of web-scale learning for capturing these subtle semantic arrangement rules. This is especially evident in the fruit scene, where DALL-E recognises the semantic connections between fruit and a fruit basket. Since it has seen many paintings and photographs of fruit in fruit baskets, it successfully predicts that this is a natural goal state. Examples of generated images used by DALL-E-Bot can be seen in Fig. 8.

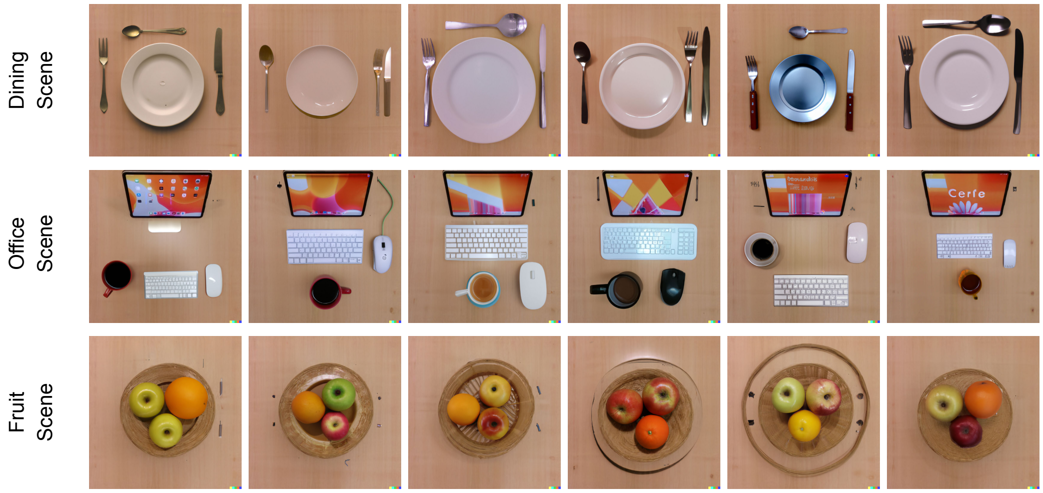

Figure 8: Examples of generated images used by DALL-E-Bot.

Out of the considered DALL-E-Bot variants, DALL-E-Bot-NF performed consistently the worst in all of the evaluation scenes. This justifies our sample-and-filter approach for using these web-scale models, rather than blindly using the first generated image. In this way, DALL-E-Bot automatically selects an arrangement which the robot can feasibly create with the objects in front of it. Examples of undesirable generations by the DALL-E-Bot-NF variant can be seen in Fig. 9.

The DALL-E-Bot-AR variant performed well in the office and fruit scenes but struggled to create human-preferable arrangements in the dining scene. The dining scene contains a larger number of thin movable objects that are more susceptible to pose estimation and execution errors. The autoregressive approach observes imperfectly placed objects and tries to place the remaining ones in a similar way. Due to this, the error accumulates in each autoregressive step resulting in an arrangement that is imprecise and less semantically correct. DALL-E-Bot avoids this issue by jointly predicting all object poses in advance.

## J.2 Quantitative Evaluation

As we can see from Table 2, the considered baselines struggle with finding correct placements for the missing object. This shows that it is challenging to design a heuristic method for this task without overfitting to one specific scene or object, and motivates our approach of learning these semantics from many images of human-made arrangements. On the other hand, DALL-E-Bot can consistently infer the preferred pose of the missing object by only observing the fixed objects. Note that each component in the pipeline will contribute to the end-to-end error, e.g. due to imperfect segmentation or pose estimation. Since our method is modular, it is easy to swap in another component, e.g. a more

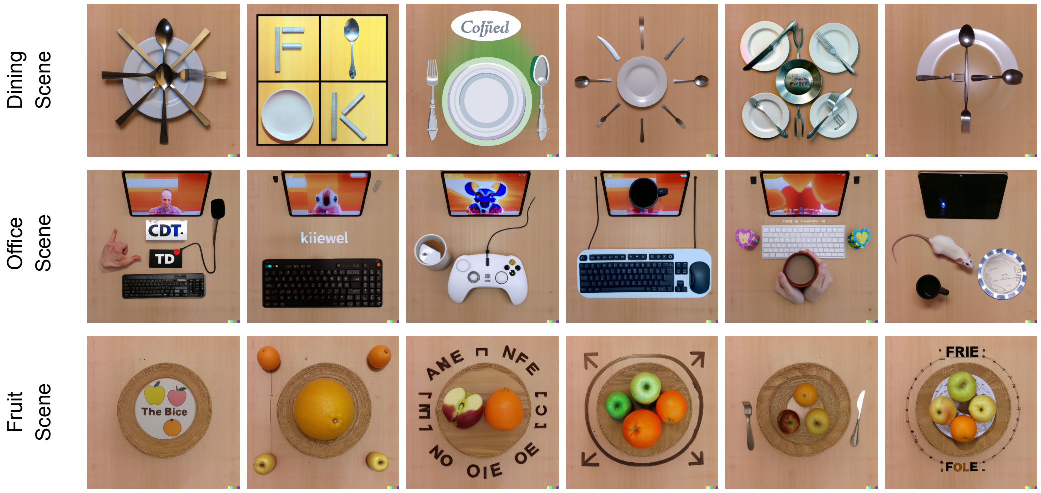

Figure 9: Examples of DALL-E generated arrangements which the robot cannot easily create with the objects that it has in the real world. Therefore, we adopt a sample-and-filter approach for using these web-scale generative models. This lets the robot choose an arrangement which is physically feasible and where the generated objects most resemble those that robot has in the real world.

powerful pose estimator if object models are available, and decrease the error in this way. Problems like instance segmentation are independent and active areas of research: as new state-of-the-art models are developed, they can easily be integrated into our method to improve its performance. Additionally, this experiment motivates the use of image generation, instead of simply searching for existing images of arranged objects. This search approach would not adequately take into account the poses of the pre-placed objects already in the scene, so is not applicable to this task. On the other hand, inpainting with diffusion models can take into account the poses of pre-placed objects, leading to a more practically useful rearrangement system.

## K Limitations & Opportunities for Future Work

Here, we discuss the limitations of this method to help researchers decide whether it is well-suited for their use-case, and propose a range of intriguing directions for future work.

**Personal preferences**. If objects placed by the user are visible in the inpainting mask, DALL-E may infer the user's implicit preferences (e.g. left/right-handedness) in order to place the remaining objects to create a coherent arrangement. However, when no objects are pre-placed by the user, then the arrangement made by the robot will likely resemble those arrangements which are commonly found in web data, and this may not align with the user's preferences. Future work could extend to conditioning on preferences inferred from previous scenes arranged by the user [10], in order to cater to the user's preferences without requiring them to begin the rearrangement themselves.

**Top-down pick-and-place**. Our experiments focus on 3-DoF rearrangement tasks, which is sufficient for many everyday tasks. However, future work can extend to 6-DoF poses with more complex interactions, e.g. to stack shelves. This could draw from recent works on collision-aware manipulation [46] and learning of skills beyond pick-and-place [47].

**Object-centric framework**. Our method reasons about pose transformations to solve everyday rearrangement tasks. Thus, as individual components (e.g. segmentation, pose estimation) improve, overall performance will also improve. However, some tasks, such as folding deformable fabrics or sweeping small particles, are not within this method's scope.

**Overlap between objects**. Currently, our method assumes that movable objects cannot overlap, so the fork cannot go on top of the plate. To handle this, the robot would need to use task planning to stack objects in the correct order. At the start of the rearrangement, the robot could spread out all the objects on the table to reduce occlusions as it detects all the objects it needs to arrange.

**Robustness of cross-domain object alignment.** We use pre-trained semantic features from ImageNet, inspired by [48], to align real and generated objects using semantic feature maps. However, the generated images sometimes lack detail: e.g. the generated keyboards lack legible text, making alignment difficult. As the quality of diffusion-generated images continues to improve, this issue will be mitigated.

**Diffusion model accessibility**. We use the public-facing interface for DALL-E from OpenAI. Although this is a paid API, there are already diffusion models such as Stable Diffusion [4] which are freely available and can be used for inference in seconds on a consumer-grade GPU. As more diffusion models become widely available, it will be feasible for any research lab or company to apply these diffusion models in their robotics setup.

**Prompt engineering**. Adding terms such as "neat, precise, ordered, geometric" for the dining scene improved the apparent neatness of the generated image. As found in other works [49], there is significant scope to explore this further. This could be used to increase the rate of semantically suitable arrangements being generated, since the desired image is more clearly specified to the diffusion model.

**Language-conditioned generation**. One exciting direction for future work is generating arrangements based on language instructions. These can easily be added to the text prompt, e.g. "plates stacked" vs "plates laid out". Generating images which match these prompts containing spatial relations may prove challenging, since prior work [50] has shown that DALL-E finds it difficult to bind textual relations to objects reliably. However, this may be overcome with future diffusion models. Note also that our method does not rely on specifying spatial relations through text, so this does not present a limitation of the current method, but this is nevertheless an important research problem for future work.

## L Recommendations to the Text-To-Image Community

As this is the first work to explore web-scale diffusion models for robotics, we now provide our findings on how future diffusion models can be made more useful for robotics.

**Everyday scenes in training datasets**. We found that Stable Diffusion [4] trained on LAION-Aesthetics is proficient at generating aesthetically pleasing images, but the DALL-E training approach may be better suited for robotic applications, because the training dataset includes a significant amount of "ordinary" images and stock photographs. Training *only* on everyday photographs could be useful.

**Visual conditioning**. Rather than just conditioning on language descriptions of objects to be generated, it would be useful to condition on image features of the real objects, but still allow the diffusion model to arrange them differently. This would help with matching between the initial and generated images. Techniques such as [51, 52] can make the generated objects better match the real instances.

**Activity-oriented datasets**. Building web-scale models which feature activities that we would like robots to perform could lead to breakthroughs in robotics. Text-to-video models [53, 54, 55] can be used as powerful world models. Even text-to-image models trained on frames from videos involving everyday activities can be useful.

**3D geometry**. Extracting 3D geometry from web-scale models trained on 2D image data [56] can allow for 6-DoF object pose estimation, making robotics methods such as DALL-E-Bot applicable to 3D scenes, e.g. stacking shelves.