

Learning Instruction-Guided Manipulation Affordance via Large Models for Embodied Robotic Tasks

Dayou Li[&], Chenkun Zhao[&], Shuo Yang, Lin Ma, Yibin Li, and Wei Zhang^{*}

Abstract—We study the task of language instruction-guided robotic manipulation, in which an embodied robot is supposed to manipulate the target objects based on the language instructions. In previous studies, the predicted manipulation regions of the target object typically do not change with specification from the language instructions, which means that the language perception and manipulation prediction are separate. However, in human behavioral patterns, the manipulation regions of the same object will change for different language instructions. In this paper, we propose Instruction-Guided Affordance Net (IGANet) for predicting affordance maps of instruction-guided robotic manipulation tasks by utilizing powerful priors from vision and language encoders pre-trained on large-scale datasets. We develop a Vision-Language-Models(VLMs)-based data augmentation pipeline, which can generate a large amount of data automatically for model training. Besides, with the help of Large-Language-Models(LLMs), actions can be effectively executed to finish the tasks defined by instructions. A series of real-world experiments revealed that our method can achieve better performance with generated data. Moreover, our model can generalize better to scenarios with unseen objects and language instructions.

I. INTRODUCTION

Humans are able to perform diverse tasks according to language instructions. Embodied robots are expected to possess “human-like” manipulation abilities in daily lives. However, it is non-trivial to endow robots with the same understanding ability and manipulation flexibility as humans. Fig. 1 shows a typical manipulation example. If you ask a person to ‘*push the coffee cup to left*’, they will manipulate the main body of the coffee cup. But when the instruction turns to ‘*pick up the cup with hot coffee*’, they tend to grab the handle of the coffee cup due to its hot temperature. This example shows that humans can select appropriate parts of an object for manipulation based on different instructions. Can we endow robots with the same capability of selecting appropriate parts of objects for instruction-guided manipulation?

Manipulation affordance indicates functional interactions of object parts with humans, which is considered an effective manipulation-centric representation for enabling diverse embodied robotic tasks. In fact, it is not a new thing to learn object manipulation affordance [1], [2]. Zeng et al. [3], [4] propose to learn action affordance by self-supervised

Dayou Li, Chenkun Zhao, Shuo Yang, Yibin Li, and Wei Zhang are with Shandong University, China. Lin Ma is with Meituan.

[&]These authors contributed equally to this work.

^{*}Corresponding author: Wei Zhang (email: davidzhang@sdu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grants 61991411 and U22A2057, and in part by the Project for Self-Developed Innovation Team of Jinan City under Grant 2021GXRC038.

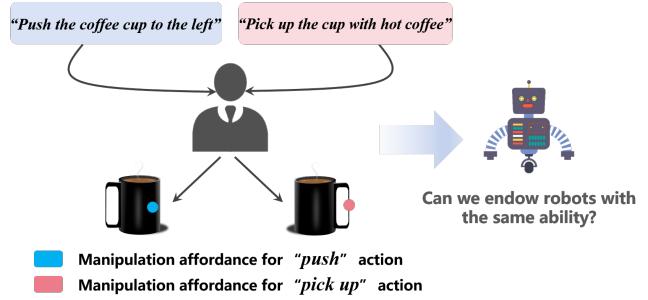


Fig. 1: Illustration of human’s reasoning process in handling instruction-guided manipulation tasks.

learning or supervised learning. The training data is obtained by agent exploring in simulation or human labeling. Lin et al. [5] attempt to directly transfer model parameters from vision models to affordance prediction networks. Mo et al. propose Where2Act [6] to predict the action affordance of 3D objects. Where2Act is capable of selecting suitable manipulation parts according to action primitives while still lacking the ability to handle language instructions. Recent progress in large language models and multi-modal models makes it easier to incorporate language instructions for manipulation affordance prediction. CLIPORT [7] integrates semantic understanding of CLIP [8] and spatial precision of Transporter [9] together into a convolutional architecture to predict manipulation affordance based on language instructions. Luo et al. [10] present a two-stage framework for grounding spatial-related instructions for affordance predictions in object manipulation tasks, which also apply CLIP to extract features of language and image input. The above methods are built upon the pre-trained CLIP model for feature extraction and then train a convolutional architecture to predict affordance maps. However, they collect data in a simulation environment or human demonstration, causing a heavy human burden and a sim2real gap.

Recently, Large Language Models (LLMs) have been employed in robotic manipulation tasks and achieved significant progress [11]–[14]. In particular, Ahn et al. [14] leveraged the probability distribution of text output of GPT-3 by calling API and combined it with value function to generate affordance value of the actions that will be probably performed in the next state. Unfortunately, this method currently only supports affordance prediction of task level.

In this paper, we aim to propose an efficient pipeline for instruction-guided robotic manipulation tasks. To this end, three core components are presented, including a manip-

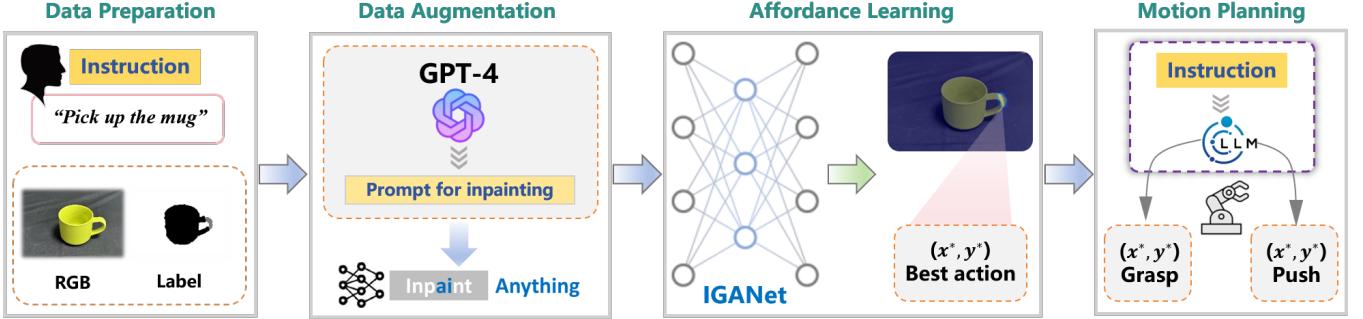


Fig. 2: Illustration of the presented full pipeline for instruction-guided manipulation tasks. Our pre-labeled dataset will be scaled up via our data augmentation pipeline. Then the IGANet is trained on the generated dataset to predict affordance maps based on the language instructions. Finally, the LLM-based planner will give commands on action execution based on the affordance maps and instructions.

ulation affordance prediction model of pixel level, a data augmentation pipeline via Vision-Language Models (VLMs), and an automatic action planner. Fig. 2 illustrates the full pipeline that integrates the three components. Specifically, we design a data augmentation pipeline based on Inpaint-Anything [15], which can edit the image based on our instruction generated by GPT-4. We manually label a small amount of real-world data and then expand the dataset through our proposed pipeline to ensure that the generated data will be similar to the real-world data. We apply a pre-trained OWL-ViT vision encoder and Universal-Sentence-Encoder to produce vision and language features for affordance prediction. Besides, an LLM-based action planner is present to determine action according to the language instructions, ensuring that the instructions can be well completed. To summarize, our main contributions are:

- We propose IGANet, an efficient framework for learning instruction-guided robotic manipulation affordance, which jointly models language and vision. An LLM-based action planner is also proposed for action planning to guide the robot through abstract instructions.
- We present an automatic data augmentation pipeline using a diffusion model as an VLM and an LLM, which can produce a large amount of data for model training.
- The proposed method is evaluated on a series of scenarios with seen and unseen objects and language instructions, demonstrating the effectiveness and generalization of our framework.

II. RELATED WORKS

A. VLMs-Driven Data Augmentation

Nowadays, Vision-Language models have shown strong capabilities in understanding image-text pairs and image generation. DALL-E, powered by OpenAI, as well as StableDiffusion [16] and Midjourney, are powerful image generators that can generate an image based on the language description. They are trained on web-scale datasets and thus have strong generalization. Many scholars apply such generative models for dataset expansion [17], [18]. Zeng et al. [18] apply VLM to generate goal rearrangement images based on the structured scene descriptions. Brooks et al. [17]

adopt GPT-3 to generate image editing instructions and use StableDiffusion [16] to generate edited images. Yu et al. [19] adopt DALL-E to scale up data for robotic manipulation task learning. Access to such generative VLMs enables us to automate the generation of large-scale datasets.

With such inspiration, we now review image editing techniques that can be used in our data augmentation pipeline. Image editing targets pixel-level editing of images for tasks such as style transfer, background replacement, image insertion, object removal, etc [20]–[24]. From Generative Adversarial Networks (GANs) to Diffusion Models, image editing methods have also been revolutionized. The sophisticated GAN-based methodology is gradually losing its dominance in image editing topic. When equipped with large-scale image-text datasets, such as Laion-5b [25] and InstructPix2Pix [17], diffusion models are endowed with powerful generative capabilities. Many image editing models such as Object 3dit [26] and InstructPix2Pix [17] have shown stable ability to edit image based on an input image and a text instruction of how to edit it. Therefore, we intend to apply this technique to expand our human-labeled small-batch data for robotic manipulation affordance.

B. Language-Guided Robotic Manipulation

Because of the availability of Large-Language Models (LLMs) and Vision-Language Models (VLMs), language-guided robotic manipulation has become a spotlight research topic in recent years. Chen et al. [27] focus on the task of grasping the target object based on a natural language command query. They adopt LSTM as a language command encoder. However, after Radford et al. released CLIP [8], the way in which language commands and images are encoded has also changed dramatically. Many scholars attempt to apply text encoder and image encoder in CLIP for their model, as CLIP is trained by a web-scale dataset. Shridhar et al. [7] present CLIPORT, which is a language-conditioned imitation learning agent that combines the broad semantic understanding of what CLIP [8] with spatial precision. Xu et al. [28] propose to jointly model vision, language, and action with object-centric representation by using both image and text encoders in CLIP. Shafiullah et al. [29] adopt

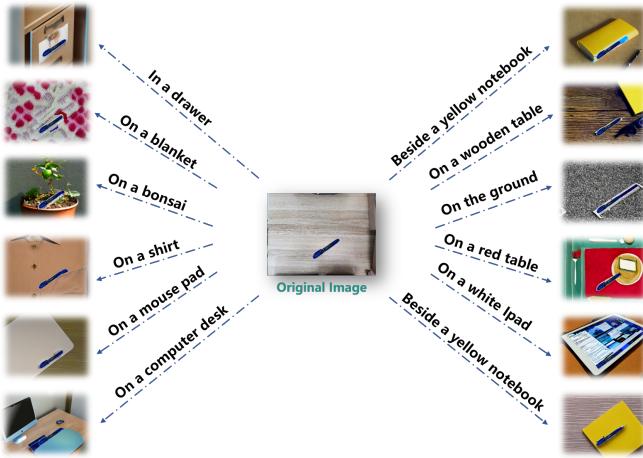


Fig. 3: Data Generation Result. Our proposed data augmentation pipeline uses GPT-4 as an LLM to generate prompts for the Inpaint-Anything module to edit the image according to the generated prompts.

CLIP embeddings to train real-world CLIP-Fields for goal navigation. Shen et al. [30] extract dense features from CLIP using the MaskCLIP reparameterization trick to support zero-shot language guidance. Rather than applying CLIP for feature extraction in language-guided manipulation, some scholars also use LLMs for action planning. Wu et al. [31] leverage the summarization capabilities of LLMs to infer generalized user preferences by planning which object should be manipulated. Sharan et al. [32] employ LLMs to generate single-step text sub-goals that will be translated into vision sub-goals via a diffusion model.

In conclusion, most pipelines that adopt LLMs as planners can only perform coarse planning for tasks. However, tasks such as fine-grained planning for object manipulation regions, usually require supervise-learning methods. However supervise-learning algorithms are in need of large-scale datasets, which are always difficult to obtain. Thus, we aim to combine the generative power of VLMs with the strong feature extraction capability of CLIP and the excellent planning capability of LLMs, in order to overcome the shortcomings respectively.

III. METHOD

A. Pipeline Overview

As shown in Fig. 2, we present Instruction-Guided Affordance Net (IGANet), a novel method for learning instruction-guided manipulation affordance. Given a prompt that defines the manipulation instruction and an image $I \in \mathbb{R}^{H \times W \times 3}$ of the target object, our objective is to produce object manipulation affordance based on the task instruction. Particularly, IGANet takes as input a text prompt denoted by p , demonstrating which object is set as the target and how it will be manipulated. The desired output of our network is an affordance map M that assigns an affordance value to each pixel of the input image, representing the operable area of the target object for the task description. Besides, we propose

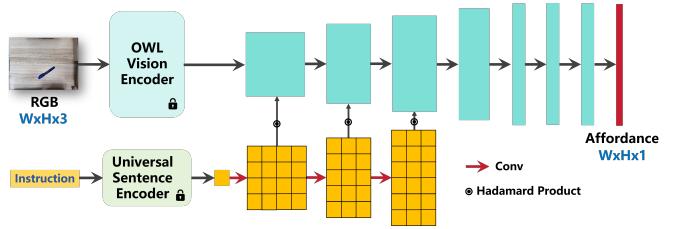


Fig. 4: Structure of IGANet. IGANet uses a frozen OWL vision encoder to encode RGB input, and the language instruction is encoded by a frozen Universal-Sentence encoder. The RGB feature and language feature perform Hadamard Product operation. The final output of IGANet is the affordance map of dense pixel-wise features.

a data augmentation pipeline using Vision-Language Models (VLMs), which is capable of generating diverse desired data automatically. Finally, with our proposed action planner, the language instructions can be broken down into executable actions, which our real-world robotic platform can perform.

B. Scaling up Data via VLMs

First, for a given object, we label different manipulation affordance according to different tasks. Then, we apply LLMs to generate some inpainting prompts that define the changes the model should make to the images. Inpaint-Anything [15] merges SAM [33], LaMA [34], and StableDiffusion [35] to enable the user to remove, fill and replace anything in the image. Inspired by Inpaint-Anything, we propose our pipeline to augment our data.

We pre-define 30 items and label their manipulation affordance with language instructions according to the task, along with their masks. We rotate the objects, and their masks, and labeled affordance in the original image to enlarge our initial dataset. Even though a hand-engineered prompt may guarantee the generated data to be out-of-distribution, the size of the generated prompts is not large enough. Therefore, we leverage common sense learned by LLMs to provide various prompts for image editing with detailed descriptions of the imagined scene. By processing the generated prompts, Inpaint-Anything can edit the image into various styles, as shown in Fig. 3. Also, we query the LLM to generate diverse language instructions for manipulation, which share similar meanings to our human-labeled data but are expressed in a different form. Using GPT-4 to generate rather than manually formulate prompts ensures their diversity. To sum up, the data format that we generate is text-image-affordance.

C. Learning Instruction-Guided Manipulation Affordance

In IGANet, as shown in Fig. 4, we propose to yield visual features through OWL-ViT [36] vision encoder denoted by E_{vit} and language features through Universal-Sentence-Encoder [37] denoted by E_{uni} . The frozen OWL-ViT vision encoder encodes RGB input to produce dense features. Then we propose a decoder consisting of several fully-connected convolutional layers and some upsample layers in between.

The instruction prompt is first encoded by the text encoder E_{uni} to produce a goal encoding $g = E_{uni}(p)$. The goal

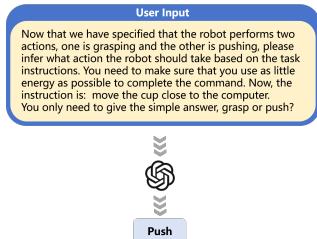


Fig. 5: LLM-Based Planner. The LLM-based planner uses GPT-4 as LLM to give action decisions based on our prompt engineering.

encoding g is then downsampled with a fully-connected layer to keep consistent with the channel dimension. Then the goal encoding is tiled to match with the spatial dimension of the decoder features. We take the Hadamard product of the decoder features and the tiled goal encodings, this element-wise product enables us to combine instruction prompt features with image features, achieving alignment between both. This instruction guiding is conducted repeatedly for three consecutive layers of the decoder. Finally, the channel dimension to 1 is reduced by applying 1 convolutional layer to the decoded features, producing manipulation affordance with ReLU function.

During training, we apply cross-entropy loss to train our model:

$$L = - \sum_{(r,c)}^{(w,h)} [P_{G(r,c)} \log P_{A(r,c)}] \quad (1)$$

where (r, c) denotes the pixel coordinate and (w, h) denotes the shape of the image. $P_{G(r,c)}$ and $P_{A(r,c)}$ represent ground truth and predicted affordance respectively.

D. Action Execution

We define two primitive actions including grasping and pushing, the following contains some specific parameters of the actions and how we propose to determine them:

- **Grasping:** A grasping action can be defined as $a_g = (p_g, \theta_g)$, where $p_g = (x_g, y_g, z_g) \in \mathbf{R}^3$ represents the middle position of the top-down parallel-jaw grasp and $\theta_g \in \mathbf{R}$ represents the grasping angle that ranges within -90° to 90° around the z-axis. We use DINO [38] to predict the target bounding box as box prompt for SAM [33] to acquire segmentation mask of the object. Then we calculate the short side tilt angle of the target's bounding box as the rotation angle. h_g is the height at point (x_g, y_g) , and $z_g = h_g - 2\text{cm}$. The gripper needs to move down 2 cm below h_g to execute the grasping action.
- **Pushing:** A pushing action can be defined as $a_p = (p_p, d_p)$ that is performed by the tip of the gripper. Each push length is fixed at 13 cm and the pushing trajectory is straight. $p_p = (x_p, y_p, z_p) \in \mathbf{R}^3$ represents the starting position of pushing action and $d_p \in \mathbf{R}^3$ is the pushing direction vector. The end position of the pushing action is derived by GPT-4 with vision (GPT-4V) based on the input language instruction. GPT-4V takes as input a description of the scene, including object categories



Fig. 6: Affordance Prediction. Here shows several visualizations of predicted affordance maps based on language instructions.

and their bounding boxes, along with the manipulation instructions and the image of the scene. Relying on the excellent detection capability of DINO, the end position of the pushing action can be roughly estimated. Thus the direction vector can be calculated.

Moreover, as shown in Fig. 5, the GPT-4V is also able to make selections of action according to the language manipulation instructions via prompt engineering.

IV. EXPERIMENTS

We conduct real-world experiments to evaluate our method. The goals of our experiments are: 1) to demonstrate that our VLM-driven data augmentation pipeline is effective in boosting the performance of our proposed model; 2) to evaluate the generalization of our model on unseen objects and language instructions; 3) to validate the effectiveness of our overall framework. Details are as follows.

A. Environment Setup

To test the real-world performance of our method, we develop an embodied robotic system to perform instruction-guided manipulation tasks. The developed system is composed of a UR5 robotic arm, an Intel RealSense D435i camera that is used to capture the observation of the workspace, and a ROBOTIQ two-finger gripper mounted on the end of the robotic arm. We set up six different scenarios with a series of objects. In each scene, we randomly select several classes of objects and create 20 different testing tasks using different instances of selected classes and diverse language instructions. A test can be judged as a success if the robot successfully manipulates the objects according to the given instructions. We report the task success rate to measure the performance of our method.

B. Baseline Methods

Given that no works adopt the same technical pipeline as us in the field of instruction-guided manipulation, we create two baseline methods based on currently popular large models. We use GPT-4 as the instruction reasoning and task planning engine as it is one of the most widely used LLMs showing powerful reasoning ability. For visual perception and manipulation frame generation, we employ two popular VLMs, OWL-ViT [36] and Grounding DINO. We term the two baseline methods GPT+ViT and GPT+DINO, respectively. To further validate the proposed VLM-driven data augmentation strategy, we additionally create a variant of

TABLE I: Real-World Experiment Results

Method	Success Rate					
	Scene 1	Scene 2	Scene 3	Scene 4	Scene 5	Scene 6
GPT + ViT	60% (12/20)	50% (10/20)	60% (12/20)	45% (9/20)	55% (11/20)	50% (10/20)
GPT + DINO	50% (10/20)	55% (11/20)	40% (8/20)	50% (10/20)	70% (14/20)	65% (13/20)
Ours w/o data augmentation	70% (14/20)	75% (15/20)	85% (17/20)	25% (5/20)	35% (7/20)	30% (6/20)
Ours	75% (15/20)	85% (17/20)	90% (18/20)	80% (16/20)	90% (18/20)	95% (19/20)

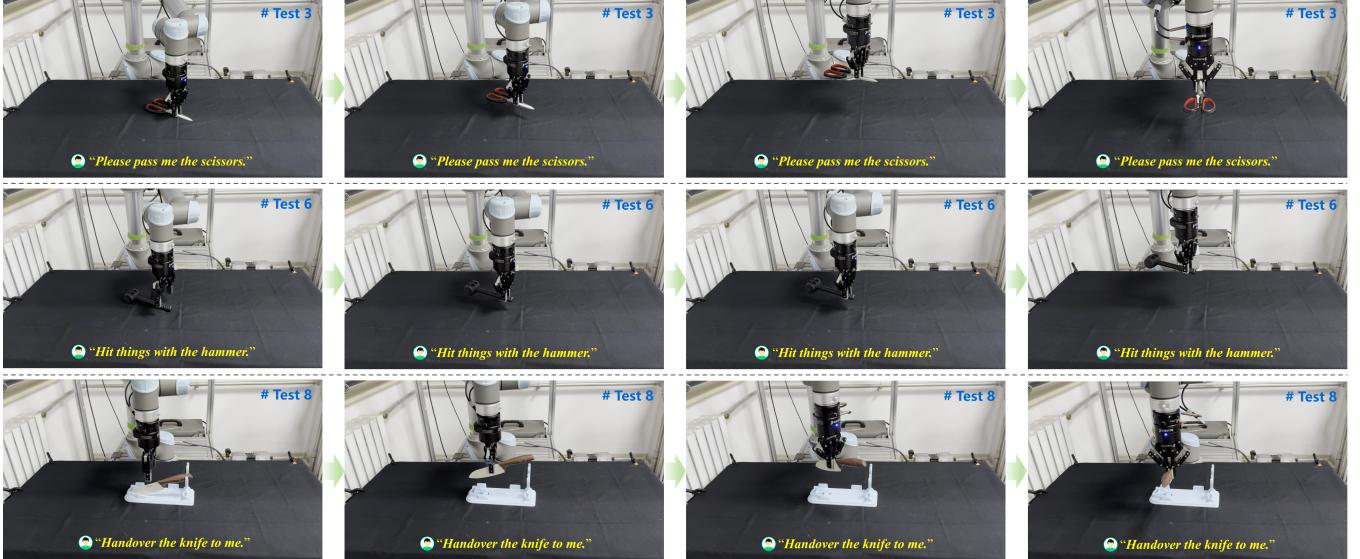


Fig. 7: Real-world Execution. Examples of the real-world test results performed by our robotic system.

our full method by removing the data augmentation, termed as Ours w/o data augmentation.

C. Results

The comparative results are provided in Table. I. The results show that our method outperforms all baseline methods by a large margin across six different scenes, validating the effectiveness of our method in performing instruction-guided manipulation tasks. The first 3 scenes contain seen objects and language instructions, while the remains contain unseen ones. The “GPT+X” methods use open-vocabulary detectors for detecting the target object and generating the center point of the bounding box as the operation position. However, from the result, we can find that such methods do not surpass us in performance, which demonstrates that determining the manipulation region by the center of the bounding box is not flexible enough to deal with versatile language instructions. From Table I, we can see that our method without our VLM-aided data generation performs badly in unseen scenarios. This is because the amount of the human-labeled data is not large enough and the model is overfitting. By training the IGANet with our generated data, we observe that the performance of our entire framework is relatively satisfactory, and our proposed VLM-based data augmentation pipeline can alleviate the loss of model performance caused by a lack of data. We provide the visualization examples of

manipulation affordance generated by IGANet in Fig. 6. It can be seen that IGANet can accurately predict different-located manipulation affordance on objects conditioned on the given instructions. Fig. 7 shows some testing cases of real-world experiments, demonstrating reliable instruction-guided manipulation performance on a real robot. Full real-world demonstrations of our method can be seen on https://youtu.be/tgQ_K1Yj2c0.

V. CONCLUSIONS

In this work, we focus on the task of instruction-guided robotic manipulation. We take full advantage of LLMs and VLMs in our proposed framework. Faced with the situation of a small amount of data, we propose a VLM-based data augmentation pipeline to generate a large amount of data automatically for model training. We apply LLMs in our action planner to assist in action execution, ensuring the actions can be performed effectively. We also utilize the pre-trained vision and language encoder in our proposed affordance prediction model (IGANet). It is worth mentioning that our method incorporates the currently popular large model technique, which provides new ideas for language-guided robotic manipulation tasks.

REFERENCES

- [1] S. Yang, W. Zhang, R. Song, J. Cheng, H. Wang, and Y. Li, “Watch and act: Learning robotic manipulation from visual demonstration,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, no. 7, pp. 4404–4416, 2023.
- [2] S. Yang, W. Zhang, R. Song, J. Cheng, and Y. Li, “Learning multi-object dense descriptor for autonomous goal-conditioned grasping,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 4109–4116, 2021.
- [3] A. Zeng, S. Song, J. Lee, A. Rodriguez, and T. Funkhouser, “Tossing-bot: Learning to throw arbitrary objects with residual physics,” *IEEE Transactions on Robotics*, vol. 36, no. 4, pp. 1307–1319, 2020.
- [4] A. Zeng, S. Song, S. Welker, J. Lee, A. Rodriguez, and T. Funkhouser, “Learning synergies between pushing and grasping with self-supervised deep reinforcement learning,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4238–4245.
- [5] L. Yen-Chen, A. Zeng, S. Song, P. Isola, and T.-Y. Lin, “Learning to see before learning to act: Visual pre-training for manipulation,” in *2020 IEEE International Conference on Robotics and Automation*, pp. 7286–7293.
- [6] 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*, pp. 6813–6823, 2021.
- [7] M. Shridhar, L. Manuelli, and D. Fox, “Cliport: What and where pathways for robotic manipulation,” in *Conference on Robot Learning*, pp. 894–906, 2022.
- [8] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., “Learning transferable visual models from natural language supervision,” in *International conference on machine learning*, pp. 8748–8763, 2021.
- [9] A. Zeng, P. Florene, J. Tompson, S. Welker, J. Chien, M. Attarian, T. Armstrong, I. Krasin, D. Duong, V. Sindhwani, et al., “Transporter networks: Rearranging the visual world for robotic manipulation,” in *Conference on Robot Learning*, pp. 726–747, 2021.
- [10] Q. Luo, Y. Li, and Y. Wu, “Grounding object relations in language-conditioned robotic manipulation with semantic-spatial reasoning,” *arXiv preprint arXiv:2303.17919*, 2023.
- [11] K. Lin, C. Agia, T. Migimatsu, M. Pavone, and J. Bohg, “Text2motion: From natural language instructions to feasible plans,” in *ICRA2023 Workshop on Pretraining for Robotics (PT4R)*, 2023.
- [12] C. H. Song, J. Wu, C. Washington, B. M. Sadler, W.-L. Chao, and Y. Su, “Llm-planner: Few-shot grounded planning for embodied agents with large language models,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2998–3009, 2023.
- [13] Y. Ding, X. Zhang, C. Paxton, and S. Zhang, “Task and motion planning with large language models for object rearrangement,” in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2086–2092.
- [14] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, C. Fu, K. Gopalakrishnan, K. Hausman, et al., “Do as i can, not as i say: Grounding language in robotic affordances,” *arXiv preprint arXiv:2204.01691*, 2022.
- [15] T. Yu, R. Feng, R. Feng, J. Liu, X. Jin, W. Zeng, and Z. Chen, “Inpaint anything: Segment anything meets image inpainting,” *arXiv preprint arXiv:2304.06790*, 2023.
- [16] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-resolution image synthesis with latent diffusion models,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.
- [17] T. Brooks, A. Holynski, and A. A. Efros, “Instructpix2pix: Learning to follow image editing instructions,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18392–18402, 2023.
- [18] Y. Zeng, M. Wu, L. Yang, J. Zhang, H. Ding, H. Cheng, and H. Dong, “Distilling functional rearrangement priors from large models,” *arXiv preprint arXiv:2312.01474*, 2023.
- [19] 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.
- [20] L. A. Gatys, A. S. Ecker, and M. Bethge, “Image style transfer using convolutional neural networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2414–2423, 2016.
- [21] A. Hertz, R. Mokady, J. Tenenbaum, K. Aberman, Y. Pritch, and D. Cohen-Or, “Prompt-to-prompt image editing with cross attention control,” *arXiv preprint arXiv:2208.01626*, 2022.
- [22] G. P. Laput, M. Dontcheva, G. Wilensky, W. Chang, A. Agarwala, J. Linder, and E. Adar, “Pixeltone: A multimodal interface for image editing,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 2185–2194, 2013.
- [23] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J.-Y. Zhu, and S. Ermon, “Sdedit: Guided image synthesis and editing with stochastic differential equations,” in *International Conference on Learning Representations*, 2021.
- [24] B. Kawar, S. Zada, O. Lang, O. Tov, H. Chang, T. Dekel, I. Mosseri, and M. Irani, “Imagic: Text-based real image editing with diffusion models,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6007–6017, 2023.
- [25] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman, et al., “Laion-5b: An open large-scale dataset for training next generation image-text models,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 25278–25294, 2022.
- [26] O. Michel, A. Bhattacharjee, E. VanderBilt, R. Krishna, A. Kembhavi, and T. Gupta, “Object 3dit: Language-guided 3d-aware image editing,” *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [27] Y. Chen, R. Xu, Y. Lin, and P. A. Vela, “A joint network for grasp detection conditioned on natural language commands,” in *2021 IEEE International Conference on Robotics and Automation*, pp. 4576–4582.
- [28] K. Xu, S. Zhao, Z. Zhou, Z. Li, H. Pi, Y. Zhu, Y. Wang, and R. Xiong, “A joint modeling of vision-language-action for target-oriented grasping in clutter,” in *2023 IEEE International Conference on Robotics and Automation*, pp. 11597–11604.
- [29] N. M. M. Shafullah, C. Paxton, L. Pinto, S. Chintala, and A. Szlam, “Clip-fields: Weakly supervised semantic fields for robotic memory,” in *ICRA2023 Workshop on Pretraining for Robotics (PT4R)*, 2023.
- [30] W. Shen, G. Yang, A. Yu, J. Wong, L. P. Kaelbling, and P. Isola, “Distilled feature fields enable few-shot language-guided manipulation,” in *Conference on Robot Learning*, pp. 405–424, 2023.
- [31] J. Wu, R. Antonova, A. Kan, M. Lepert, A. Zeng, S. Song, J. Bohg, S. Rusinkiewicz, and T. Funkhouser, “Tidybot: Personalized robot assistance with large language models,” in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3546–3553.
- [32] S. Sharan, R. Zhao, Z. Wang, S. P. Chinchali, et al., “Plan diffuser: Grounding llm planners with diffusion models for robotic manipulation,” in *Bridging the Gap between Cognitive Science and Robot Learning in the Real World: Progresses and New Directions*, 2024.
- [33] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, et al., “Segment anything,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4015–4026, 2023.
- [34] R. Suvorov, E. Logacheva, A. Mashikhin, A. Remizova, A. Ashukha, A. Silvestrov, N. Kong, H. Goka, K. Park, and V. Lempitsky, “Resolution-robust large mask inpainting with fourier convolutions,” in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 2149–2159, 2022.
- [35] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-resolution image synthesis with latent diffusion models,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.
- [36] M. Minderer, A. Gritsenko, A. Stone, M. Neumann, D. Weissenborn, A. Dosovitskiy, A. Mahendran, A. Arnab, M. Dehghani, Z. Shen, et al., “Simple open-vocabulary object detection,” in *European Conference on Computer Vision*, pp. 728–755, 2022.
- [37] D. Cer, Y. Yang, S.-y. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, et al., “Universal sentence encoder for english,” in *Proceedings of the 2018 conference on empirical methods in natural language processing: system demonstrations*, pp. 169–174.
- [38] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, C. Li, J. Yang, H. Su, J. Zhu, et al., “Grounding dino: Marrying dino with grounded pre-training for open-set object detection,” *arXiv preprint arXiv:2303.05499*, 2023.