LAN-grasp: An Effective Approach to Semantic Object Grasping Using Large Language Models

Reihaneh Mirjalili[1], Michael Krawez[1], Simone Silenzi[1], Yannik Blei[1] and Wolfram Burgard[1]

***Abstract*— In this paper, we propose LAN-grasp, a novel approach towards more appropriate semantic grasping. We use foundation models to provide the robot with a deeper understanding of the objects, the right place to grasp an object, or even the parts to avoid. This allows our robot to grasp and utilize objects in a more meaningful and safe manner. We leverage the combination of a Large Language Model, a Vision Language Model, and a traditional grasp planner to generate grasps demonstrating a deeper semantic understanding of the objects. We first prompt the Large Language Model about which object part is appropriate for grasping. Next, the Vision Language Model identifies the corresponding part in the object image. Finally, we generate grasp proposals in the region proposed by the Vision Language Model. Building on foundation models provides us with a zero-shot grasp method that can handle a wide range of objects without the need for further training or fine-tuning. We evaluated our method in real-world experiments on a custom object data set. We present the results of a survey that asks the participants to choose an object part appropriate for grasping. The results show that the grasps generated by our method are consistently ranked higher by the participants than those generated by a conventional grasping planner and a recent semantic grasping approach.**

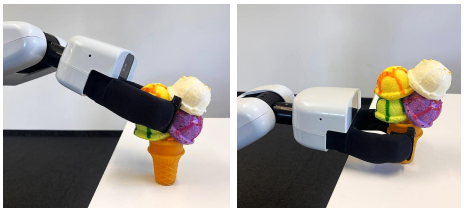
I. Introduction

Objects found in household environments often require a specific way of interaction which ensures various criteria such as avoiding to damage the object, user safety, functionality, etc. As robots are increasingly involved in human living environments, it is crucial to provide them with sufficient semantic knowledge about these environments and the objects found within them.

Traditional approaches to robotic grasping [1], [2], [3] only analyze the object geometry and aim to optimize the grasp stability. Recent data-driven approaches [4], [5], [6] also account for the object class and can generate grasps appropriate for the specific object type. However, most of these methods require substantial computational resources for training and can fail to generalize to unseen object categories. Our objective is an approach for object-specific grasping that ensures tool usability and safety without any need for further training.

We proceed towards this goal by introducing LAN-grasp, a zero-shot method built on foundation models. The scale of these models and the massive size and generality of their training data allow us to reason about a large variety of objects without further training or fine-tuning. In particular, LAN-grasp uses a Large Language Model (LLM) to understand which part of an object is suitable for grasping.

[1]All authors are with the Department of Engineering, University of Technology Nuremberg, Germany.



Graspit! LAN-grasp

Fig. 1: Robot performing the command of *"Pick up the ice cream please"*. The grasp on the left is generated without including semantic information while the grasp on the right is performed using our method leveraging a deeper object understanding provided by Large Language Models.

Next, this information is grounded in the object image by leveraging a Vision Language Model (VLM). Our method uses GPT-4 as LLM and OWL-Vit [10] as VLM. However, due to the modular structure of LAN-grasp, it can easily be adapted to use other LLMs or VLMs. Finally, we use an off-the-shelf grasp proposal tool [2] to plan the grasps in accordance with the admissible parts of the object detected by the deployed foundation models[2].

In summary, we make the following contributions:

1) We propose a novel approach using foundation models for zero-shot semantic object grasping.
2) We demonstrate that the presented approach can work with a wide variety of day-to-day objects without the need for additional training.
3) We evaluate our approach by asking human participants to choose the appropriate grasps.

II. Related Work

Traditional grasping algorithms [1], [2], [3], [11] analyze the geometry of the object and the gripper to propose and evaluate a grasping pose. Building on decades of development, these methods are fast and reliable off-the-shelf tools. However, they do not incorporate semantic information and operate based on object shape only. Also, such methods rely on a precise object model and thus suffer from partial or noisy geometry. Data-driven approaches regress grasping candidates from either single view RGB images [4], [12] or point clouds [13], [14], thus mitigating the need for a complete object model. Further, a network can learn a more natural grasping policy if human-like grasps are included in the training data, where such grasps are either created manually [15] or learned through imitation [6]. Do *et al.* [17]

[2]Video available at https://tinyurl.com/5bnwpkuc.

propose an end-to-end trained network that detects object instances in an image and assigns pixel-wise affordance masks to object parts. Monica and Aleotti [19] propose a system that decomposes an object point cloud into meaningful parts which then serve as grasping targets. However, the part the robot has to grasp is provided by the user whereas in our method the part is suggested by an LLM.

Recently, foundation models have attracted a lot of attention in different sub-fields of robotics [20], [21], [22]. Ngyen *et al.* [23] train an open-vocabulary affordance detector for point clouds whereby CLIP is deployed to encode the affordance labels. Similarly, Tang *et al.* [8] use CLIP to facilitate task-specific grasping from RGB images and language instructions. Song *et al.* [25] use BERT as the language back-end and train a network that grounds object parts in a point cloud from a user instruction. Here, however, the part label is explicitly referred to in the user input. The approach of Tang *et al.* [9] lifts this limitation by prompting an LLM to describe the shape and parts of an object. The LLM response is then processed by a Transformer-based grasp evaluation network. Our method also relies on an LLM for deciding what object part should be grasped. The crucial difference to the above works is that our approach relies solely on foundation models and does not require any training. Thus, once more powerful foundation models are available, the performance of our approach is easily improved by switching to a novel LLM or VLM.

III. Method Description

In this section, we explain the details of LAN-grasp. The pipeline consists of two main parts: the *Language Module* and the *Grasp Planning Module*. The overview of our approach is depicted in Figure 2.

A. Language Module

In the first step, the object label `<object>` provided by the user is transferred into a LLM prompt in the following format:

```
"role": "system", "content": "You are an
intelligent robotic arm."
"role": "user", "content": "If you want
to pick up an <object>, which part makes
the most sense to grasp? Name one part."
```

The scheme of the prompt is chosen to be compatible with GPT-4 which is the LLM that we used in the pipeline [28]. We included the last sentence to prevent the LLM from giving extra explanations and thus only output the desired object part. We use OWL-Vit [10] as the VLM for grounding the object part label in the image. It builds on the Vision Transformer Architecture, first presented by Dosovitskiy *et al.* [29]. OWL-Vit detects and marks the desired object part with a bounding box which is projected on the object 3D model.

B. Grasp Planning Module

We deploy the GraspIt! simulator [32] as our grasp proposal generator. It is a standard tool that operates on geometric models and evaluates grasps according to physical constraints. Thus, the first step for grasp planning is to create a dense 3D mesh model of the object. In our setup, we use two fixed RGB-D sensors and a turning table for object scanning. We acquire the camera poses from an Aruco board and integrate the depth images via KinectFusion [33].

To generate feasible grasps, GraspIt! splits the scene into object and obstacle geometry, and we exploit this mechanism by marking the mesh parts that project into the VLM-generated bounding box as object and the rest as obstacle. This enforces grasping only at the desired object part. The resulting grasp proposals are ranked based on grasp efficiency and finger friction. In case the object part suggested by the LLM is not detected in the image our system considers the full object geometry, i.e., it falls back to the vanilla grasp planner.

We want to point out that our approach is agnostic about the grasp planner and could be potentially replaced by other tools that do not require a complete object model.

IV. Experimental Evaluation

In this section, we present the details of our experiments and results. Our goal is to demonstrate that our method proposes to grasp object parts that are preferred by humans on a variety of objects. We argue that humans generally choose grasps that enable correct tool usage and ensure safety and that a robot retains these desirable qualities by executing similar grasps. To that end, we first collect a data set of typical household objects. Next, we apply our approach to these objects and execute the grasping on a real robot. Finally, we show that our grasping strategy is similar to human preferences obtained through a survey and that our approach outperforms two baselines based on this similarity metric.

A. Dataset

We collect a data set containing 22 different objects commonly found in household environments. We chose these objects to cover a wide range of situations where semantic knowledge is required for proper grasping. Our first objective was to showcase grasping on functional objects like tools or kitchen supplies, e.g., *shovel*, *hand brush*, and *knife*. Further, we included delicate objects that might be damaged with an improper grasp, for instance, *rose*, *cupcake*, and *ice cream*. For other objects, a wrong grasp can cause a dangerous situation, e.g., *candle*. Finally, we include objects where an improper grasp might not necessarily be harmful but is rather unnatural to a human observer, for instance, *doll*, *bag*, and *wine glass*. The objects in the data set are shown in Figure 1, Figure 3, and Figure 4.

B. Experimental Setup and Baselines

Our first baseline is the plain GraspIt! simulator. Here we use the same 3D models as for our approach but do not restrict grasping to the object part selected by the language model. The second baseline is GraspGPT [9], a recent approach to task-oriented grasping (ToG). Though our

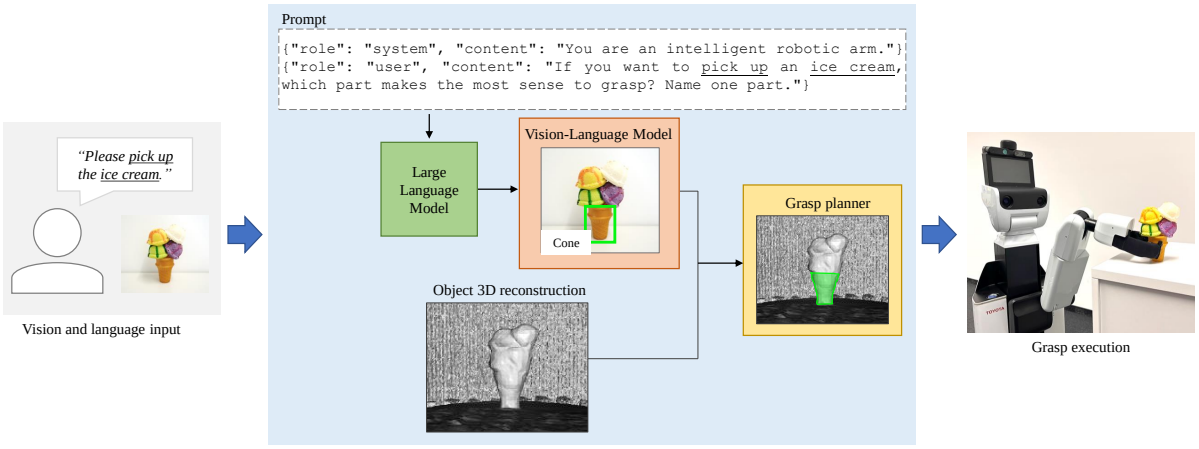


Fig. 2: Our approach in a nutshell: First, the command from the user is turned into a prompt suitable for the LLM. Next, the LLM outputs the proper part for grasping the object, which in this example is the *cone*. This label is then grounded in the object image using a VLM. The object 3D model and the object part mask are then passed to the grasp planner, which is restricted to generate grasps only in the masked region. The best grasp candidate is finally executed by the robot.

method performs semantic grasping and not ToG, we chose this baseline because it also leverages an LLM similar to ours. GraspGPT requires as input an object point cloud and a natural language prompt describing the object, the object class, and the task. We generate the point clouds from the object meshes reconstructed as above and use an object-specific activity as the task label, for instance, "to drink" for a mug. Further, we experiment with different task labels as GraspGPT input for each object and only report the best baseline results according to our evaluation metric. Also we note that GraspGPT uses GPT-3 as LLM back-end, whereas we rely on GPT-4. However, from preliminary experiments with GPT-3 we found that for the task at hand the differences to GPT-4 are negligible. All real-world experiments are executed on the Human Support Robot (HSR) [35] as shown in Figure 3.

C. Qualitative Results

The grasps executed on the HSR are shown in Figure 3. For the rest of the objects, the grasping area proposed by our method is visualized in Figure 4. The results suggest that LAN-grasp proposes grasps suitable for the usage of the respective object. For instance, grasping the *handle* for *shovel* and *broom* corresponds to the intended use of these items. For *lollipop* and *cupcake*, the grasp is placed away from the edible part at the *stick* and the *wrapper*, respectively. It is noteworthy that our method is able to understand the relation between stacked objects, e.g., *flowers in a vase* or *plate of cake*. Also, for a single *cup*, LAN-grasp suggests grasping the *handle* while for the *cup on a saucer* the grasp proposal is the *saucer*. Other objects, e.g., *doll*, *bag*, or *wine glass*, do not possess a critical area where grasping would cause harm or directly interfere with the functionality. However, our method is able to generate grasps that are closer to how a human would handle these items. In contrast to LAN-grasp, the areas suggested by GraspIt! are expectantly random and do not consider semantic intricacies.

TABLE I: Similarity of grasping area preferences compared to a human user. The left half of the table lists the objects and the object part the majority of survey participants suggested for grasping, with the corresponding percentage of users. The right half of the table shows the similarity scores per object for the two baselines and our proposed method.

Object	Preferred Part	GraspIt!	GraspGPT	LAN-grasp
doll	torso 92.1%	0.28	0.48	**0.92**
ice cream	cone 100.0%	0.05	0.40	**1.00**
candle	base 93.1%	0.22	0.57	**0.93**
flowers in the vase	vase 93.2%	0.32	0.73	**0.93**
bag	handle 91.1%	0.79	0.69	**0.91**
plant	pot 94.3%	0.16	0.56	**0.94**
hand brush	handle 95.4%	0.65	**0.95**	**0.95**
toilet brush	handle 97.6%	0.42	0.52	**0.98**
cactus	pot 98.8%	0.26	**0.99**	**0.99**
cupcake	wrapper 100.0%	0.10	0.40	**1.00**
cup on a saucer	saucer 81.2%	0.24	0.59	**0.81**
plate of cake	plate 98.8%	0.11	0.51	**0.99**
mug	handle 77.1%	0.28	0.73	**0.77**
saucepan	handle 94.3%	0.36	**0.94**	**0.94**
broom	handle 97.6%	0.42	**0.98**	**0.98**
Average		0.31	0.67	**0.94**

D. Quantitative Results

To support the claim that our approach proposes grasps similar to human preferences, we designed a questionnaire on grasping choices. A group of 83 participants were presented with images of all objects used in the experiments and were asked where they would grasp them. For each object, the participants could choose between two parts marked by bounding boxes in the image. The survey results are summarized in Table I. Per object, we state the preferred part and the percentage of participants that selected it.

Next, we evaluate how similar the generated grasps are compared to the ones suggested by human users. Given that

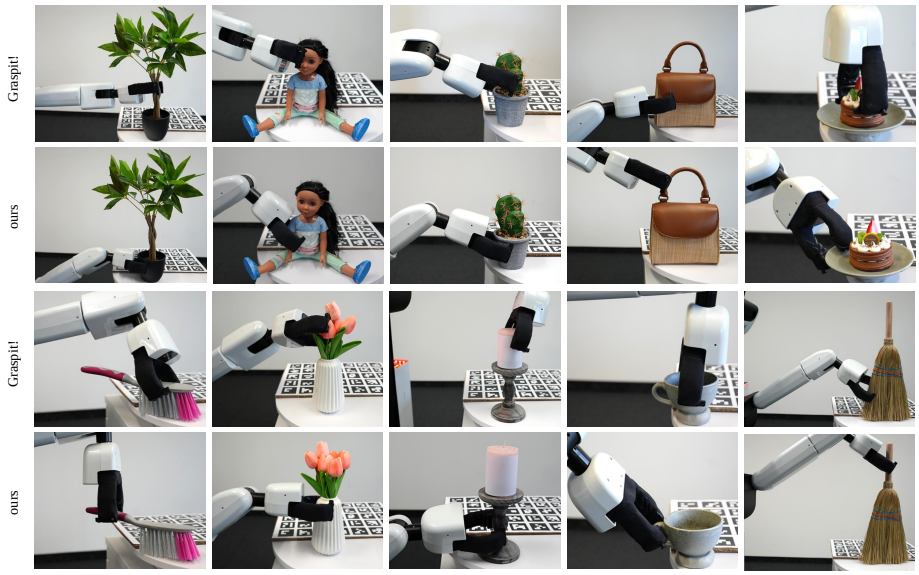


Fig. 3: The performed grasps by the HSR robot: Each column presents the grasps for one object. The first row for each object, shows the grasp generated without semantic knowledge about the objects, while the second row shows the grasps generated by LAN-grasp.

Fig. 4: The results of LAN-grasp on a set of common household objects. The green bounding box shows the area to grasp suggested by our method.

an object is segmented into parts a and b, let $p_a \in [0, 1]$ be the empirical probability that a method grasps at part a and $p_b = 1 - p_a$ that part b is grasped. Further, let p_a^h be the human grasping frequency at a according to the survey results and p_a^x the corresponding frequency produced by one of the considered methods. To compute p_a^x for the baselines, we obtained the best 20 grasp proposals from each algorithm and counted the grasps falling into region a. LAN-grasp restricts the grasps to the object part selected by the LLM, which in our experiments robustly proposed the same part for a given object. Thus, the values of p_a^x were here either 1 or 0. Finally, we computed a per-object similarity score for each method x as $sim_x = 1 - |p_a^h - p_a^x|$. These scores are shown in Table I along with the average similarity scores over all objects.

Our method consistently outperforms the baselines on the similarity score and ties only in four cases with GraspGPT. The average similarity score of LAN-grasp is considerably higher with the value of 0.94 compared to 0.31 achieved by GraspIt! and 0.67 achieved by GraspGPT. We further note that in all cases, the object part choice of LAN-grasp coincides with the majority vote of the survey participants. The low score of GraspIt! is not surprising since it only considers geometric and not semantic aspects of the object. GraspGPT exhibits a better performance compared to GraspIt! due to leveraging semantic concepts and LLMs. GraspGPT is trained on a data set mostly containing tools and house supplies and thus performs best on objects close to its training data distribution. However, the performance drops on objects outside of its training data like a *doll* or an *ice cream*.

We do not evaluate grasping stability, since our focus is on choosing a reasonable object part to grasp. In future, a grasp stability measure could be included into the grasp candidate selection.

E. Conclusion and Future Work

In this paper, we presented LAN-grasp, a novel approach to semantic object grasping. By leveraging foundation models, we provide our approach with a deep understanding of the objects and their intended use in a zero-shot manner. Through extensive experiments, we showed that for a wide range of objects LAN-grasp is generating grasps that are preferred by humans and also ensure safety and object usability. In particular, the proposed grasps were compared to human preferences gathered through a questionnaire. The evaluation showed that LAN-grasp performs consistently better on that metric than the baseline methods. Inspired by these results, in future we plan to further exploit LLMs to not only decide where to grasp an object but also how to grasp and hold it according to a specific task.

REFERENCES

[1] A. Bicchi, "On the closure properties of robotic grasping," *International Journal of Robotics Research (IJRR)*, 1995.

[2] A. T. Miller and P. K. Allen, "Graspit!: A versatile simulator for grasp analysis," in *ASME International Mechanical Engineering Congress and Exposition*, 2000.

[3] A. Ten Pas, M. Gualtieri, K. Saenko, and R. Platt, "Grasp pose detection in point clouds," *International Journal of Robotics Research (IJRR)*, 2017.

[4] E. Jang, S. Vijayanarasimhan, P. Pastor, J. Ibarz, and S. Levine, "End-to-end learning of semantic grasping," *arXiv preprint arXiv:1707.01932*, 2017.

[5] J. H. Kwak, J. Lee, J. J. Whang, and S. Jo, "Semantic grasping via a knowledge graph of robotic manipulation: A graph representation learning approach," *IEEE Robotics and Automation Letters (RA-L)*, 2022.

[6] Y.-H. Wu, J. Wang, and X. Wang, "Learning generalizable dexterous manipulation from human grasp affordance," in *Conference on Robot Learning (CoRL)*, 2023.

[7] A. Murali, W. Liu, K. Marino, S. Chernova, and A. Gupta, "Same object, different grasps: Data and semantic knowledge for task-oriented grasping," in *Conference on Robot Learning (CoRL)*, 2021.

[8] C. Tang, D. Huang, L. Meng, W. Liu, and H. Zhang, "Task-oriented grasp prediction with visual-language inputs," *arXiv preprint arXiv:2302.14355*, 2023.

[9] C. Tang, D. Huang, W. Ge, W. Liu, and H. Zhang, "Graspgpt: Leveraging semantic knowledge from a large language model for task-oriented grasping," *arXiv preprint arXiv:2307.13204*, 2023.

[10] 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 (ECCV)*, 2022.

[11] B. S. Zapata-Impata, P. Gil, J. Pomares, and F. Torres, "Fast geometry-based computation of grasping points on three-dimensional point clouds," *International Journal of Advanced Robotic Systems*, 2019.

[12] J. Redmon and A. Angelova, "Real-time grasp detection using convolutional neural networks," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[13] B. Zhao, H. Zhang, X. Lan, H. Wang, Z. Tian, and N. Zheng, "Regnet: Region-based grasp network for end-to-end grasp detection in point clouds," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

[14] A. Alliegro, M. Rudorfer, F. Frattin, A. Leonardis, and T. Tommasi, "End-to-end learning to grasp via sampling from object point clouds," *IEEE Robotics and Automation Letters (RA-L)*, 2022.

[15] E. Corona, A. Pumarola, G. Alenya, F. Moreno-Noguer, and G. Rogez, "Ganhand: Predicting human grasp affordances in multi-object scenes," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[16] Y. Chen, R. Xu, Y. Lin, and P. A. Vela, "A joint network for grasp detection conditioned on natural language commands," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

[17] T.-T. Do, A. Nguyen, and I. Reid, "Affordancenet: An end-to-end deep learning approach for object affordance detection," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018.

[18] W. Liu, A. Daruna, and S. Chernova, "Cage: Context-aware grasping engine," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2020.

[19] R. Monica and J. Aleotti, "Point cloud projective analysis for part-based grasp planning," *IEEE Robotics and Automation Letters (RA-L)*, 2020.

[20] R. Mirjalili, M. Krawez, and W. Burgard, "Fm-loc: Using foundation models for improved vision-based localization," *arXiv preprint arXiv:2304.07058*, 2023.

[21] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Audio visual language maps for robot navigation," *arXiv preprint arXiv:2303.07522*, 2023.

[22] Y. Cui, S. Karamcheti, R. Palleti, N. Shivakumar, P. Liang, and D. Sadigh, "No, to the right: Online language corrections for robotic manipulation via shared autonomy," in *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*, 2023.

[23] T. Ngyen, M. N. Vu, A. Vuong, D. Nguyen, T. Vo, N. Le, and A. Nguyen, "Open-vocabulary affordance detection in 3d point clouds," *arXiv preprint arXiv:2303.02401*, 2023.

[24] J. Gao, B. Sarkar, F. Xia, T. Xiao, J. Wu, B. Ichter, A. Majumdar, and D. Sadigh, "Physically grounded vision-language models for robotic manipulation," *arXiv preprint arXiv:2309.02561*, 2023.

[25] Y. Song, P. Sun, Y. Ren, Y. Zheng, and Y. Zhang, "Learning 6-dof fine-grained grasp detection based on part affordance grounding," *arXiv preprint arXiv:2301.11564*, 2023.

[26] D. Driess, F. Xia, M. S. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu *et al.*, "Palm-e: An embodied multimodal language model," *arXiv preprint arXiv:2303.03378*, 2023.

[27] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn *et al.*, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," *arXiv preprint arXiv:2307.15818*, 2023.

[28] OpenAI, "Gpt-4 technical report," *arXiv:2303.08774*, 2023.

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

[30] X. Zhai, X. Wang, B. Mustafa, A. Steiner, D. Keysers, A. Kolesnikov, and L. Beyer, "Lit: Zero-shot transfer with locked-image text tuning," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[31] C. Jia, Y. Yang, Y. Xia, Y.-T. Chen, Z. Parekh, H. Pham, Q. Le, Y.-H. Sung, Z. Li, and T. Duerig, "Scaling up visual and vision-language representation learning with noisy text supervision," in *International Conference on Machine Learning (ICML)*, 2021.

[32] A. T. Miller and P. K. Allen, "Graspit! a versatile simulator for robotic grasping," *IEEE Robotics & Automation Magazine (RAM)*, 2004.

[33] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2011.

[34] A. T. Miller, S. Knoop, H. I. Christensen, and P. K. Allen, "Automatic grasp planning using shape primitives," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2003.

[35] T. Yamamoto, K. Terada, A. Ochiai, F. Saito, Y. Asahara, and K. Murase, "Development of human support robot as the research platform of a domestic mobile manipulator," *ROBOMECH journal*, 2019.