Few-Shot Learning for Language-Driven Grasp Detection

Le Nam Khanh
namkhanh2172@gmail.com

Abstract

To address the challenge of few-shot learning in language-driven grasp detection, we introduce GraspLLaMA. This approach enhances data robustness through augmentation and leverages transfer learning to effectively train on limited datasets. The source code is open-sourced at:https://github.com/khanhkhanhlele/GraspDetection

1. Introduction

Grasp detection is a cornerstone of robotics, essential for enabling robots to interact with their environments effectively. The integration of advanced large language models (LLMs) has revolutionized this field, allowing robots to understand and execute complex human commands through natural language. Despite these advancements, challenges persist in training these models with limited data. To address this, we introduce GraspLLaVA, a model that optimally leverages pre-trained components and data augmentation to excel in few-shot learning scenarios, thereby facilitating more intuitive and robust human-robot interactions. Our contribution can be summarized as follows:

- We developed a model that integrates LLM into robotic grasping techniques, so called vision language model (VLM).
- We utilized transfer learning by fine-tuning a pre-trained model and employed data augmentation techniques to improve training effectiveness on few-shot datasets while preserve the high computation overheads induced by training LLMs.

2. Related Work

Grasp detection relies on object geometry [1]. With the rise of deep learning, data-driven methods [2] became more common. Redmon, *et al.* [7] pioneered the use of deep neural networks to predict multiple grasp candidates for a single object. Other method [3] perform grasp detection on depth images as input, where accuracy is highly dependent on the quality of the input data.

3. Problem Formulation

We address the problem of learning a mapping function $\mathbf{F}$ that receives a visual scene observation $\mathbf{O} \in \mathbb{R}^{H \times W \times 3}$ and a task instruction $\mathbf{I} = \{\mathbf{s}_t\}_{t=1}^{T}$, and outputs a task-oriented grasp pose $\mathbf{g} \in \mathbb{R}^5$ in the image space. $\mathbf{s}_t$, T represents the t-th word token and the maximum length of the instruction, respectively. We have $\mathbf{g} = \mathbf{F}(\mathbf{O}, \mathbf{I})$ where $\mathbf{O}$ is an RGB image. $\mathbf{I}$ is a natural language sentence describing the task. Figure 1 shows two examples of the output grasp poses. Each grasp pose $\mathbf{g_i}$ is a 5-dimensional rectangle defined by the grasp location (x_i, y_i), orientation θ_i, opening width w_i, and length h_i.

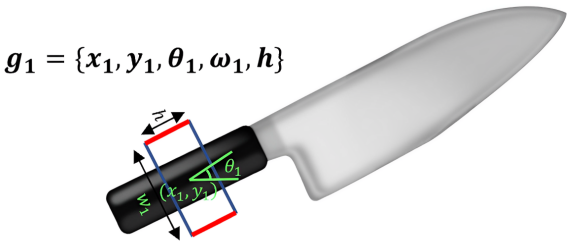

Figure 1. Example of grasp poses as parameterized by location, orientation, width, and length.

4. Method

An overview of the proposed LLaVA is presented in Fig. 2. Task instructions $\mathbf{I}$, and the image object $\mathbf{O}$, are processed through a VLM to produce textual output. Subsequently, we extract the embeddings for all corresponding tokens. The embeddings are computed and then passed through MLP layer. Simultaneously, the image object $\mathbf{O}$ is processed through an encoder to extract visual features, which are then concatenated with the embeddings obtained from the reasoning module to output $\mathbf{v_{enc}}$. The decoder predicts a task-oriented grasp pose based on $\mathbf{v_{enc}}$ and return quality head $\mathbf{M}_q \in \mathbb{R}^{H \times W}$, orientation map $\mathbf{M}_\theta \in \mathbb{R}^{H \times W \times 2}$ and opening width map $\mathbf{M}_w \in \mathbb{R}^{H \times W}$, respectively.

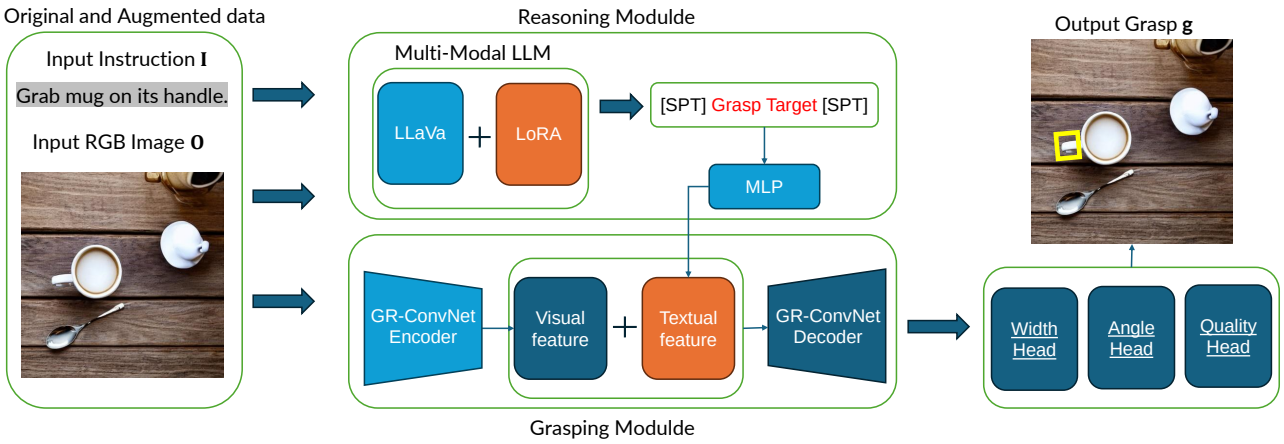

Figure 2. An Overview of LLaVA architecture. The model uses visual images **O** and textual instructions **I** to generate the grasp pose **g** for a specific target object or part. The grasp target embeddings are then forwarded to the grasping module for pose detection.

4.1. Preprocessing

Images are rotated and their bounding boxes are recalibrated to enhance accuracy. Textual inputs are also paraphrased using a LLM to diversify language understanding, ensuring the dataset is robust for real-world applications.

4.2. Architecture

The model architecture consists of 2 major components: The reasoning module and the grasping module.

Reasoning module. We employ the pre-trained LLaVA [6], which is augmented with LoRA [4] fine-tuning. LLaVA combines a CLIP visual encoder with LLaMA. Additionally, LoRA recognized for its computational efficiency, is integrated into the projection layer as well as all the linear layers of the LLMs.

Grasping module. We utilize GR-ConvNet [5], which has been pre-trained on the Cornell Grasping Dataset, facilitating the application of transfer learning to scenarios involving few-shot data.

4.3. Loss function

The loss function consists of a location $\mathbf{M}_q$, an orientation $\mathbf{M}_\theta$, and an opening width $\mathbf{M}_w$:

$$\mathcal{L} = \mathcal{L}_{\text{loc}}(\mathbf{M}_q, \hat{\mathbf{M}}_q) + \beta_1 \mathcal{L}_{\text{ori}}(\mathbf{M}_\theta, \hat{\mathbf{M}}_\theta) + \beta_2 \mathcal{L}_{\text{wid}}(\mathbf{M}_w, \hat{\mathbf{M}}_w)$$

where $\mathcal{L}_{\text{loc}}, \mathcal{L}_{\text{ori}}, \mathcal{L}_{\text{wid}}$ are the empirical risks, characterized by MSE. β_1, β_2 are the scaling coefficients. $\hat{\mathbf{M}}_q$, $\hat{\mathbf{M}}_\theta$, and $\hat{\mathbf{M}}_w$ are ground truth of $\mathbf{M}_q, \mathbf{M}_\theta, \mathbf{M}_w$, respectively.

5. Experimental Evaluations

Table 1 presents the grasp prediction outcomes on the **GraspAnything++** dataset using the Jacquard index. A grasp is considered correct if the angle discrepancy between the predicted and actual grasp is within 30°, and their Intersection over Union (IoU) exceeds 0.25.

Table 1. Performance on the GraspAnything++ dataset.

Method	Seen	Unseen
GraspLLaVA	33.69%	11.51%

6. Conclusion

This paper presents a new approach to Language-driven grasp detection under limited data conditions. Utilizing the reasoning capabilities of a multi-modal LLM, our method interprets indirect verbal commands to generate relevant grasping poses. Experimental results on the **GraspAnything++** dataset demonstrate high performance.

References

[1] A. Bicchi and V. Kumar. Robotic grasping and contact: A review. In *IEEE Int. Conf. Robot. Autom.*, 2000. 1

[2] A. Bohg, J. a Morales, T. Asfour, and D. Kragic. Data-driven grasp synthesis—a survey. *IEEE Trans. Robot.*, 2013. 1

[3] N. Gkanatsios, G. Chalvatzaki, P. Maragos, and J. Peters. Orientation attentive robot grasp synthesis. *arXiv preprint arXiv:2006.05123*, 2020. 1

[4] E. J. Hu, Y. Shen, P. Wallis, Z. A. Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 2

[5] S. Kumra, S. Joshi, and F. Sahin. Antipodal robotic grasping using generative residual convolutional neural network. In *IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2020. 2

[6] H. Liu, C. Li, Q. Wu, and Y. J. Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. 2

[7] J. Redmon and A. Angelova. Real-time grasp detection using convolutional neural networks. In *IEEE Int. Conf. Robot. Autom.*, 2015. 1