Lightweight Language-driven Grasp Detection using Conditional Consistency Model

Nghia Nguyen[1], Minh Nhat Vu[2,3,*], Baoru Huang[4], An Vuong[1], Ngan Le[5], Thieu Vo[6], Anh Nguyen[7]

***Abstract*— Language-driven grasp detection is a fundamental yet challenging task in robotics with various industrial applications. In this work, we present a new approach for language-driven grasp detection that leverages the concept of lightweight diffusion models to achieve fast inference time. By integrating diffusion processes with grasping prompts in natural language, our method can effectively encode visual and textual information, enabling more accurate and versatile grasp positioning that aligns well with the text query. To overcome the long inference time problem in diffusion models, we leverage the image and text features as the condition in the consistency model to reduce the number of denoising timesteps during inference. The intensive experimental results show that our method outperforms other recent grasp detection methods and lightweight diffusion models by a clear margin. We further validate our method in real-world robotic experiments to demonstrate its fast inference time capability.**

I. INTRODUCTION

Grasping is one of the fundamental tasks in robotics, enabling robots to interact with the physical world through a broad spectrum of applications, from industrial automation and human-robot interaction to service robotics [1]. Recent advancements in machine vision have significantly improved the capabilities of grasp detection for the robot [2]–[6]. Prior research has demonstrated encouraging grasp detection results in both 2D images [4], [7] and 3D point clouds [8], [9]. However, most existing works define grasp detection as a region localization problem while ignoring the use of natural language to localize possible grasps on the object based on linguistic input [10].

With the recent advances in Large Language Models (LLM), integrating language to robotic systems has become more popular [11]. Pretrained models such as ChatGPT [12] and CLIP [13] have revolutionized various applications and their adaptability to the robotic domain has shown encouraging results [14]–[17]. Although there are several language-driven robotic manipulations works, most focus on understanding high-level actions and overlook the fundamental grasping task [18]. In this paper, we tackle the *language-driven grasp detection* task that allows the robot to grasp specific objects based on the language command. With the language-driven grasping ability, the robot would be able to interact more effectively with the surrounding environment and humans.

Compared to the traditional grasp detection task without text, language-driven grasping offers several advantages. Firstly, we communicate with robots by providing language prompts that direct them to execute precise tasks [6], [10], [14]–[17]; therefore, the incorporation of natural language instructions augments robotic systems with the ability to interactively respond to dynamic, real-time tasks [19]. Secondly, the utilization of natural language addresses the challenge of ambiguity in identifying target objects within cluttered environments [20] or distinguishing among objects with similar shapes [21]. Lastly, linguistic guidance enriches robotic systems with semantic information [22], enhancing their learning capabilities without necessitating expert demonstrations or specific engineering [23].

Recently, several works on grasp detection have utilized diffusion models as the key technique and shown encouraging results [15], [24], [25]. This is motivated by the proven efficacy of diffusion models in conditional generation tasks [26] such as image synthesis, image segmentation, and visual grounding [24]. The effectiveness of diffusion models comes from their iterative approach to gradually refine data from an initial state of pure noise toward a meaningful output. Nonetheless, applying diffusion models to language-driven tasks in robotics faces a key challenge, *i.e.*, the inference time of diffusion models is usually not fast enough for real-time robotic applications. Consequently, recent studies have introduced techniques to tackle the inference speed problem of diffusion models using approaches such as rapid sampling [27]–[29], knowledge distillation [29], [30], or model optimization [31], [32]. However, these models are still unable to perform fast sampling with language conditions during inference to meet the real-time requirement in robotic grasping.

In this paper, we propose a new lightweight diffusion model to tackle the inference speed problem in utilizing the diffusion model for the language-driven grasp detection task. To this end, we exploit the capabilities of flow-based generative models to improve the precision of robots in identifying grasp poses from textual inputs. In particular, we develop a conditional consistency model for fast inference speed for real-time robotic applications. We verify our proposed method on a recent large-scale language-driven grasping dataset and achieve superior results in both accuracy and inference speed compared with recent approaches. Furthermore, our method enables zero-shot learning and generalize to real-world robotic grasping applications.

[1] FPT Software AI Center, Vietnam nghiant100@fpt.com
[2] Automation & Control Institute, TU Wien, Vienna, Austria
[3] Austrian Institute of Technology (AIT) GmbH, Austria
[4] Imperial College London, UK
[5] University of Arkansas, USA
[6] National University of Singapore, Singapore
[7] Department of Computer Science, University of Liverpool, UK
[*] Corresponding author minh.vu@ait.ac.at

Our contributions are summarized as follows:

- We present Lightweight Language-driven Grasp Detection (LLGD), a fast diffusion model for language-driven grasp detection.
- We conduct intensive analysis to validate our method and demonstrate that it outperforms other approaches in terms of both accuracy and execution speed.

II. Related Work

Grasp Detection. Grasp detection has been a central topic in robotics, aiming to equip robots with the ability to identify and execute object grasping in complex environments [4]–[6], [33]–[35]. Several works such as that by Redmon *et al.* [36] have set the foundation for robot grasping by using convolutional neural networks (CNNs). Most previous grasp detection methods are often limited to simple tasks with a fixed number of classes and rely solely on raw image data [5], [7], [36]. Several works [9], [37], [38] have extended the problem by using RGB-D images or 3D point clouds to output the results in 3D space. However, they still have not focused on integrating language as the input instruction in the grasp detection problem.

Language-driven Grasping. Language-driven grasp detection introduces the use of natural language to inform grasp detection tasks [6], [15], [39]–[41]. The common approach to tackling the task of language-driven grasp detection is to divide it into a two-step process. One stage is dedicated to identifying the target object, and the second stage focuses on generating grasp poses based on the established visual-text correlations [33]. Foundation models such as Ground-DINO [42], CLIP [13] have emerged, enabling zero-shot detection and zero-shot segmentation. These models allow for the localization of the target object without training [17]. However, due to their large size, they result in longer inference times. Accessing such commercial foundation models is not always possible, especially since LLM models often require the use of APIs which come at a high cost.

Lightweight Diffusion Model. Lightweight diffusion models that maintain performance while reducing computational overhead have become crucial in machine learning. Habibian *et al.* [30] utilized knowledge distillation for low-resolution features to reduce the number of parameters in U-Net. Song *et al.* [43] introduced the concept of score-based generative models. Recently, consistency models have surfaced as a strong approach of generative models capable of producing high-quality images within a single or a limited number of steps [29]. Although there are significant applications in generative tasks, these models are mostly *unconditional* [28], [29], [31]. On the other hand, robotic applications remain discriminative, making the use of unconditional diffusion models not entirely suitable. In this study, we address this issue by building a lightweight diffusion with *language conditions*. We aim to enhance the consistency model work [29] to inherit its fast inference time while adding the language conditions to make it more suitable for the language-driven grasping task.

III. Lightweight Language-driven Grasp Detection

A. Overview

Given an input RGB image and a text prompt describing the object of interest, we aim to detect the grasping pose on the image that best matches the text prompt input. We follow the popular *rectangle grasp* convention widely used in previous work to define the grasp [4], [5], [44]. We represent the target grasp pose as $\mathbf{x}_0$ in the diffusion model. The objective of our diffusion process of language-driven grasp detection involves denoising from a noisy state $\mathbf{x}_T$ to the original grasp pose $\mathbf{x}_0$, conditioned on the input image and grasp instruction represented by y. The forward process in traditional conditional diffusion model [26] is defined as:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}) \ , \tag{1}$$

where the hyperparameter β_t is the amount of noise added at diffusion step $t \in [0, T] \subset \mathbb{R}$.

To train a diffusion model with condition y, we use a neural network to learn the reverse process:

$$p_\phi(\mathbf{x}_{t-1}|\mathbf{x}_t, y) = \mathcal{N}(\mu_\phi(\mathbf{x}_t, t, y), \Sigma_\phi(\mathbf{x}_t, t, y)) \ . \tag{2}$$

In our approach, we utilize the diffusion process in the continuous domain, where $\mathbf{x}_t$ is the grasp pose state at arbitrary time index t. Unlike popular discrete diffusion models as in previous studies [24], [26], [45]–[47], by using a continuous space, we can improve sample quality and reduce inference times due to the ability to traverse the diffusion process at arbitrary timesteps, allowing for more fine-grained control over the denoising process [31].

B. Conditional Consistency Model for LLGD

To reduce the inference time during the denoising step of the diffusion model, we aim to estimate the original grasp pose with just a few denoising steps. Since our language-driven grasp detection task has the condition y, we introduce a *conditional* consistency model based on the consistency concept in [29] to infer the original grasp pose during the inference process directly:

$$\mathbf{f}_\theta(\mathbf{x}_t, t, y) = \begin{cases} \mathbf{x}_t & t \in [0, \epsilon] \\ \mathbf{F}_\theta(\mathbf{x}_t, t, y) & t \in (\epsilon, T] \end{cases} \ , \tag{3}$$

where $\mathbf{f}_\theta(\mathbf{x}_\epsilon, t, y) = \mathbf{x}_\epsilon$ is the boundary condition, $\mathbf{F}_\theta(\mathbf{x}_t, t, y)$ is a free-form deep neural network whose output has same dimentionality as $\mathbf{x}_t$.

To train our conditional consistency model, we employ knowledge distillation from a continuous diffusion process:

$$d\mathbf{x}_t = -\frac{1}{2}\gamma_t\mathbf{x}_t dt + \sqrt{\gamma_t}d\mathbf{w}_t \ , \tag{4}$$

where γ_t is non-negative function referred as noise schedule, $\mathbf{w}_t$ is the standard Brownian motion [31]. This forward process creates a trajectory of grasp pose $\{\mathbf{x}_t\}_{t=0}^T$. The grasp pose state $\mathbf{x}_t$ is not only dependent on the time index t but also on the input image and text prompt. The grasp distribution $p(\mathbf{x}_0|y)$ from dataset is transformed into

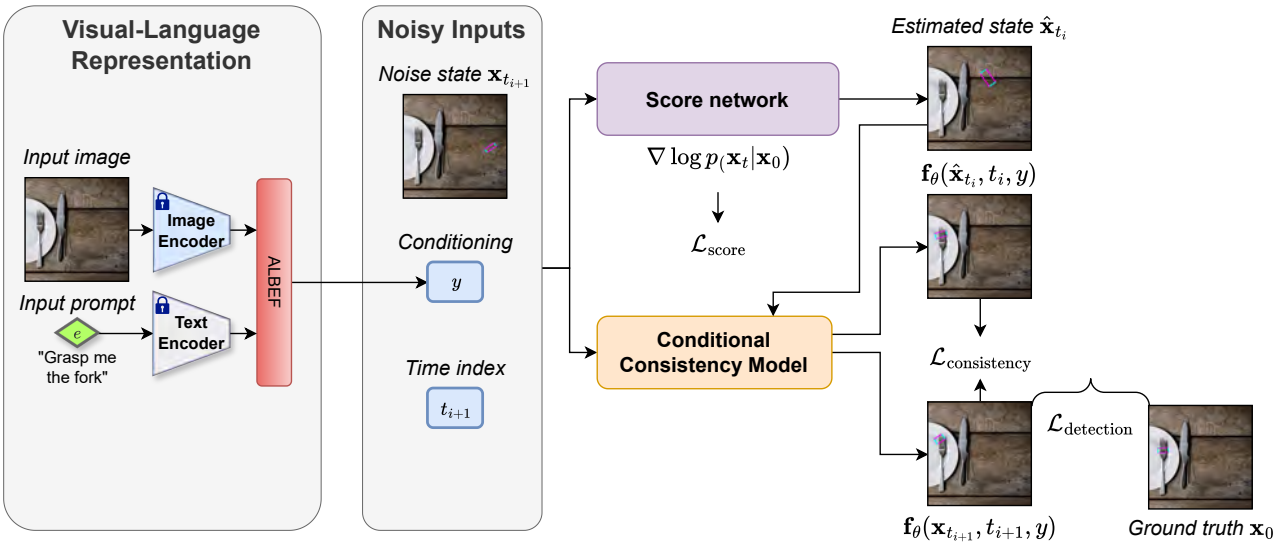

Fig. 1. The overview of our method. First, the input RGB image and text prompt are fed into the feature encoder and ALBEF fusion [48]. Subsequently, we concurrently train two models with the same architectures: A score network to estimate the probability flow Ordinary Differential Equation (ODE) trajectory [43] for the diffusion process and a conditional consistency model to determine the grasp pose with a few denoising steps.

$p(\mathbf{x}_T|y) \sim \mathcal{N}(0, \mathbf{I})$. Given ground truth grasp pose $\mathbf{x}_0$, we can sample $\mathbf{x}_t$ at arbitrary t:

$$p(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mu_t, \Sigma_t) \ , \tag{5}$$

where

$$\mu_t = e^{\frac{1}{2}\rho_t}\mathbf{x}_0, \Sigma_t = (1 - e^{\rho_t})\mathbf{I}, \rho_t = -\int_0^t \gamma_s ds \ .$$

Equation 4 is a probability flow ODE [43]. With the conditional variable y, it can be redefined as:

$$\frac{d\mathbf{x}_t}{dt} = -\frac{1}{2}\gamma_t \left[\mathbf{x}_t + \nabla \log p(\mathbf{x}_t|y)\right] \ , \tag{6}$$

where $\nabla \log p(\mathbf{x}_t|y)$ is score function of conditional diffusion model.

Suppose that we have a neural network $\mathbf{s}_\phi(\mathbf{x}_t, t, y)$ that can approximate the score function $\nabla \log p(\mathbf{x}_t|y)$, *i.e.*, $\mathbf{s}_\phi(\mathbf{x}_t, t, y) \approx \nabla \log p(\mathbf{x}_t|y)$, after training the score network, we can replace the $\nabla \log p(\mathbf{x}_t|y)$ term in Equation 6 with a neural network:

$$\frac{d\mathbf{x}_t}{dt} = -\frac{1}{2}\gamma_t \left[\mathbf{x}_t + \mathbf{s}_\phi(\mathbf{x}_t, t, y)\right] \ . \tag{7}$$

Score Function Loss. In order to approximate score function $\nabla \log p(\mathbf{x}_t|y)$, the conditional denoising estimator minimizes following objective:

$$\mathcal{L}_{\text{score}} = \mathbb{E}_{\substack{t\sim\mathcal{U}[0,T] \\ \mathbf{x}_0,y\sim p(\mathbf{x}_0,y) \\ \mathbf{x}_t\sim p(\mathbf{x}_t|\mathbf{x}_0)}} \left[\lambda(t)\|\nabla \log p(\mathbf{x}_t|\mathbf{x}_0) - \mathbf{s}_\phi(\mathbf{x}_t, t, y)\|^2\right] \ , \tag{8}$$

where $\lambda(t) \in \mathbb{R}^+$ is a positive weighting function.

Proposition 1. *Suppose that $\mathbf{x}_t$ is conditionally independent of y given $\mathbf{x}_0$, then minimizing of $\mathcal{L}_{\text{score}}$ is the same as minimizing:*

$$\mathbb{E}_{\substack{t\sim\mathcal{U}[0,T] \\ \mathbf{x}_t,y\sim p(\mathbf{x}_t,y)}} \left[\lambda(t)\|\nabla \log p(\mathbf{x}_t|y) - \mathbf{s}_\phi(\mathbf{x}_t, t, y)\|^2\right] \ .$$

Proof: Because $\mathbf{x}_t$ is conditionally independent of y given $\mathbf{x}_0$, we have

$$\begin{aligned}
&\mathbb{E}_{\substack{t\sim\mathcal{U}[0,T] \\ \mathbf{x}_0,y\sim p(\mathbf{x}_0,y) \\ \mathbf{x}_t\sim p(\mathbf{x}_t|\mathbf{x}_0)}} \left[\lambda(t)\|\nabla \log p(\mathbf{x}_t|\mathbf{x}_0) - \mathbf{s}_\phi(\mathbf{x}_t, t, y)\|^2\right] \\
&= \mathbb{E}_{\substack{t\sim\mathcal{U}[0,T] \\ y\sim p(y) \\ \mathbf{x}_0\sim p(\mathbf{x}_0|y) \\ \mathbf{x}_t\sim p(\mathbf{x}_t|\mathbf{x}_0)}} \left[\lambda(t)\|\nabla \log p(\mathbf{x}_t|\mathbf{x}_0) - \mathbf{s}_\phi(\mathbf{x}_t, t, y)\|^2\right] \\
&= \mathbb{E}_{\substack{t\sim\mathcal{U}[0,T] \\ y\sim p(y) \\ \mathbf{x}_0\sim p(\mathbf{x}_0|y) \\ \mathbf{x}_t\sim p(\mathbf{x}_t|\mathbf{x}_0,y)}} \left[\lambda(t)\|\nabla \log p(\mathbf{x}_t|\mathbf{x}_0, y) - \mathbf{s}_\phi(\mathbf{x}_t, t, y)\|^2\right] \\
&= \mathbb{E}_{\substack{t\sim\mathcal{U}[0,T] \\ y\sim p(y)}} \left[\Phi(t, y)\right] \ ,
\end{aligned} \tag{9}$$

where

$$\begin{aligned}
&\Phi(t, y) \\
&= \mathbb{E}_{\substack{\mathbf{x}_0\sim p(\mathbf{x}_0|y) \\ \mathbf{x}_t\sim p(\mathbf{x}_t|\mathbf{x}_0,y)}} \left[\lambda(t)\|\nabla \log p(\mathbf{x}_t|\mathbf{x}_0, y) - \mathbf{s}_\phi(\mathbf{x}_t, t, y)\|^2\right] \ .
\end{aligned}$$

If y and t are fixed, we can define a transition probability not depend on these variables, $q(\mathbf{x}_0) = p(\mathbf{x}_0|y)$, $\kappa(\mathbf{x}_t) = \mathbf{s}_\phi(\mathbf{x}_t, t, y)$. According to [49], we have:

$$\begin{aligned}
\Phi(t, y) &= \mathbb{E}_{\substack{\mathbf{x}_0\sim q(\mathbf{x}_0) \\ \mathbf{x}_t\sim q(\mathbf{x}_t|\mathbf{x}_0)}} \left[\lambda(t)\|\nabla \log q(\mathbf{x}_t|\mathbf{x}_0) - \kappa(\mathbf{x}_t)\|^2\right] \\
&= \mathbb{E}_{(\mathbf{x}_0,\mathbf{x}_t)\sim q(\mathbf{x}_0,\mathbf{x}_t)} \left[\lambda(t)\|\nabla \log q(\mathbf{x}_t|\mathbf{x}_0) - \kappa(\mathbf{x}_t)\|^2\right] \\
&= \mathbb{E}_{\mathbf{x}_t\sim q(\mathbf{x}_t)} \left[\lambda(t)\|\nabla \log q(\mathbf{x}_t) - \kappa(\mathbf{x}_t)\|^2\right] \\
&= \mathbb{E}_{\mathbf{x}_t\sim p(\mathbf{x}_t|y)} \left[\lambda(t)\|\nabla \log p(\mathbf{x}_t|y) - \mathbf{s}_\phi(\mathbf{x}_t, t, y)\|^2\right] \ .
\end{aligned} \tag{10}$$

From Equation 9 and 10, we can prove the equivalence of the two objective functions.

$$
\begin{aligned}
&\mathbb{E}_{\substack{t\sim\mathcal{U}[0,T]\\ \mathbf{x}_0,y\sim p(\mathbf{x}_0,y)\\ \mathbf{x}_t\sim p(\mathbf{x}_t|\mathbf{x}_0)}}\left[\lambda(t)\|\nabla\log p(\mathbf{x}_t|\mathbf{x}_0)-\mathbf{s}_\phi(\mathbf{x}_t,t,y)\|^2\right]\\
&=\mathbb{E}_{\substack{t\sim\mathcal{U}[0,T]\\ y\sim p(y)\\ \mathbf{x}_t\sim p(\mathbf{x}_t|y)}}\left[\lambda(t)\|\nabla\log p(\mathbf{x}_t|y)-\mathbf{s}_\phi(\mathbf{x}_t,t,y)\|^2\right]\\
&=\mathbb{E}_{\substack{t\sim\mathcal{U}[0,T]\\ (\mathbf{x}_t,y)\sim p(\mathbf{x}_t,y)}}\left[\lambda(t)\|\nabla\log p(\mathbf{x}_t|y)-\mathbf{s}_\phi(\mathbf{x}_t,t,y)\|^2\right]\ .
\end{aligned}
\tag{11}
$$

■

Discretization. Consider discretizing the time horizon $[\epsilon, T]$ into $N-1$ with boundary $t_1=\epsilon<t_2<t_3<\ldots<t_N=T$. If N is sufficiently large, we can use an ODE-solver [50] to estimate the next discretization step:

$$
\begin{aligned}
\hat{\mathbf{x}}_{t_i}&=\mathbf{x}_{t_{i+1}}+(t_i-t_{i+1})\left.\frac{d\mathbf{x}}{dt}\right|_{t=t_{i+1}}\\
&=\mathbf{x}_{t_{i+1}}-\frac{1}{2}\gamma_{i+1}(t_i-t_{i+1})\left[\mathbf{x}_{t_{i+1}}+\mathbf{s}_\phi(\mathbf{x}_t,t,y)\right]\ .
\end{aligned}
\tag{12}
$$

Conditional Consistency Model Loss. To enable fast sampling, we expect that the predicted point $\hat{\mathbf{x}}_{t_i}$ and $\mathbf{x}_{t_{i+1}}$ to lie on the same probability flow ODE trajectory. We propose conditional consistency lost to enforce this constraint:

$$
\begin{aligned}
\mathcal{L}_{\text{consistency}}=\mathbb{E}_{\substack{i\sim\mathcal{U}[1,N-1]\\ \mathbf{x}_{t_{i+1}}\sim p(\mathbf{x}_{t_{i+1}}|\mathbf{x}_0)}}&\\
\left[\lambda(t_i)\|\mathbf{f}_\theta(\mathbf{x}_{t_{i+1}},t_{i+1},y)-\mathbf{f}_{\theta^*}(\hat{\mathbf{x}}_{t_i},t_i,y)\|^2\right]&,
\end{aligned}
\tag{13}
$$

where $\hat{\mathbf{x}}_{t_i}$ is calculated in Equation 12, $\mathbf{x}_{t_{i+1}}$ is sampling from Gaussian distribution in Equation 5, θ is parameters of neural network $\mathbf{f}$.

Additionally, we need to minimize the discrepancy between the predicted and ground truth grasp poses with the detection loss:

$$
\mathcal{L}_{\text{detection}}=\mathbb{E}_{\substack{i\sim\mathcal{U}[1,N]\\ \mathbf{x}_{t_i}\sim\mathcal{N}(\mu_{t_i},\Sigma_{t_i})\\ \mathbf{x}_0,y\sim p(\mathbf{x}_0,y)}}\left[\lambda(t_i)\|\mathbf{f}_\theta(\mathbf{x}_{t_i},t_i,y)-\mathbf{x}_0\|^2\right]\ .
\tag{14}
$$

The overall training objective for our method is:

$$
\mathcal{L}_{\text{total}}=\mathcal{L}_{\text{consistency}}+\mathcal{L}_{\text{detection}}\ .
\tag{15}
$$

C. Network Details

The input of our network is the image and a corresponding grasping text prompt represented as e (for example, "grasp the fork at its handle"). We first extract the image feature using a 12-layer vision transformer ViT [51] image encoder. The input text prompt is encoded by a text encoder using BERT [52] or CLIP [13]. We then combine and learn the features of the input text prompt and input image using the ALBEF fusion network [48]. The output of the fusion features is fed into a score network and our conditional consistency model to learn the grasp pose. Figure 1 shows the detail of our network.

Score Network. In practice, we utilize a score network composed of several MLP layers to extract three components: the noisy grasp pose $\mathbf{x}_t$, the time index t, and the conditional vision-language embedding y. Subsequently, these features are concatenated and the score function is extracted through a final MLP layer. It is crucial to ensure that the output dimension of the score network is identical to the dimension of the input $\mathbf{x}_t$ because, fundamentally, the score function is the gradient of the grasp pose distribution given the condition y. Our conditional consistency model's network has an architecture similar to the score network; however, its output is the predicted grasp pose.

Algorithm 1 Inference Process

Input: Image and text prompt, conditional consistency model $\mathbf{f}_\theta(\mathbf{x},t,y)$, number of inference step P, sequence of time points $t_1=\epsilon<t_2<t_3<\cdots<t_P=T$, noise scheduler $\alpha_t=e^{\rho_t}$.

$y\leftarrow$ ALBEF (image, prompt)
Initial grasp noise $\mathbf{x}_T\sim\mathcal{N}(0,\mathbf{I})$
$\mathbf{x}_0\leftarrow\mathbf{f}_\theta(\mathbf{x}_T,T,y)$
for $i=P-1$ **to** 2 **do**
  Sample $\mathbf{z}\ \sim\mathcal{N}(0,\mathbf{I})$
  $\mathbf{x}_{t_i}\leftarrow\sqrt{\alpha_{t_i}}\mathbf{x}_0+\sqrt{1-\alpha_{t_i}}\mathbf{z}$
  $\mathbf{x}_0\leftarrow\mathbf{f}_\theta(\mathbf{x}_{t_i},t_i,y)$
end
Output: Final grasp pose $\mathbf{x}_0$

D. Training and Inference

During the training, we freeze the text encoder and image encoder, and then train the ALBEF fusion, the score network, and the consistency model end-to-end. We note that the score network and the conditional consistency model share the same architecture. We trained both models simultaneously for 1000 epochs with a batch size of 8 using Adam optimizer. The training time takes approximately 3 days on an NVIDIA A100 GPU. Regarding the parameters of the conditional consistency model, we empirically set $T=1000$, $\epsilon=1$, and $N=2000$. After training the score network and the conditional consistency model $\mathbf{f}_\theta(\mathbf{x}_t,t,y)$, we can sample the grasp pose given the input image and language instruction prompt in a few denoising steps using our Algorithm 1.

IV. Experiments

A. Experiment Setup

Dataset. We use the Grasp-Anything dataset [33] in our experiment. Grasp-Anything is a large-scale dataset for language-driven grasp detection with 1M samples. Each image in the dataset is accompanied by one or several prompts describing a general object grasping action or grasping an object at a specific location.

Evaluation Metrics. Our primary evaluation metric is the success rate, defined similarly to [33], [44], necessitating an IoU score of the predicted grasp exceeding 25% with the ground truth grasp and an offset angle less than 30°. We also use the harmonic mean ('H') to measure the overall success rates as in [53]. The latency (inference time) in seconds of all methods is reported using the same NVIDIA A100 GPU.

B. Comparison with Grasp Detection Methods

TABLE I
COMPARISION WITH TRADITIONAL GRASP DETECTION METHODS

Baseline	Seen	Unseen	H	Latency
GR-ConvNet [44]	0.37	0.18	0.24	**0.022**
Det-Seg-Refine [4]	0.30	0.15	0.20	0.200
GG-CNN [54]	0.12	0.08	0.10	0.040
CLIPORT [6]	0.36	0.26	0.29	0.131
CLIP-Fusion [15]	0.40	0.29	0.33	0.157
MaskGrasp [41]	0.50	0.46	0.45	0.116
LLGD (ours) with 1 timestep	0.47	0.34	0.40	0.035
LLGD (ours) with 3 timesteps	0.52	0.38	0.45	0.106
LLGD (ours) with 10 timesteps	**0.53**	**0.39**	**0.46**	0.264

We compare our LLGD with GR-CNN [44], Det-Seg-Refine [4], GG-CNN [54], CLIPORT [6], MaskGrasp [41], and CLIP-Fusion [15]. Table I compares our method and other baselines on the GraspAnything dataset. This table shows that our proposed LLGD outperforms traditional grasp detection methods by a clear margin. Our inference time is also competitive with other methods.

C. Comparison with Lightweight Diffusion Models

TABLE II
COMPARISON WITH DIFFUSION MODELS FOR LANGUAGE-DRIVEN GRASP DETECTION

Method	Seen	Unseen	H	Latency
LGD [40] with 3 timesteps	0.42	0.29	0.35	0.074
LGD [40] with 30 timesteps	0.49	0.41	0.45	0.741
LGD [40] with 1000 timesteps	0.52	0.42	0.47	26.12
SnapFusion [27] with 500 timesteps	0.49	0.37	0.43	12.95
LightGrad [31] with 250 timesteps	0.51	0.34	0.43	6.420
LLGD (ours) with 1 timestep	0.47	0.34	0.40	**0.035**
LLGD (ours) with 3 timesteps	0.52	0.38	0.45	0.106
LLGD (ours) with 10 timesteps	**0.53**	**0.39**	**0.46**	0.264

In this experiment, we compare our LLGD with other diffusion models for language-driven grasp detection. In particular, we compare with LGD [40] using DDPM [26], and recent lightweight diffusion works: SnapFusion [27] with 500 timesteps and LightGrad [31] with 250 timesteps.

Table II shows the result diffusion models for language-driven grasp detection. We can see that the accuracy and inference time of the classical diffusion model LGD strongly depend on the number of denoising timesteps. LGD with 1000 timesteps achieves reasonable accuracy but has a significant long latency. Lightweight diffusion models such as SnapFusion [27] and LightGrad [31] show reasonable results and inference speed. However, our method achieves the highest accuracy with the fastest inference speed.

D. Conditional Consistency Model Demonstration

In this analysis, we will verify the effectiveness of our conditional consistency model. In Figure 2, we visualize grasp pose aspect to time index t. In the LGD [40] model, as the discrete diffusion model is employed with $T = 1000$, we have to perform the diffusion steps with a step size of 1,

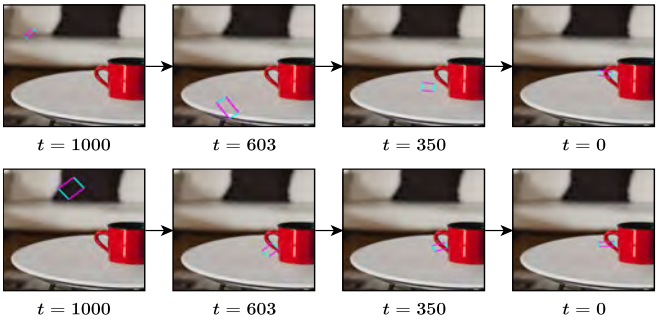

Fig. 2. Consistency model analysis. With text prompt input *"Grasp the cup at its handle"*, we compare the trajectory grasp pose of our method and LGD [40]. In the figure, the top row illustrates the trajectory of LGD, while the bottom row corresponds to the trajectory of our LLGD.

which results in very slow inference speed. Moreover, the grasp pose trajectory still exhibits significant fluctuations. Our method can arbitrarily select boundary time points for the continuous consistency model. It is evident that the number of iterations required by our method is significantly less than that of LGD [40] for the same value of T, which contributes to the "lightweight" factor. Furthermore, the grasp pose at $t = 603$ has almost converged to the ground truth, while LGD [40] using DDPM at $t = 350$ has not yet achieved a successful grasp.

E. Ablation Study

Visualization. Figure 3 shows qualitative results of our method and other baselines. The outcomes suggest that our method LLGD generates more semantically plausible grasp poses given the same text query than other baselines. In particular, other methods usually show grasp poses at the location that is not well-aligned with the text query, while our method shows more suitable detection results.

In the Wild Detection. Figure 4 illustrates the outcomes of applying our method to random images from the internet. The results demonstrate that our LLGD can effectively detect the grasp pose given the language instructions on real-world images. Our method showcases a promising zero-shot learning ability, as it successfully interprets grasp actions on images it has never encountered during training.

Failure Cases. Although good results have been achieved, our method still predicts incorrect grasp poses. A large number of objects and grasping prompts pose a challenging problem as the network cannot capture all the diverse circumstances that arise in real life. Figure 5 depicts some failure cases where LLGD incorrectly predicts the results, which can be attributed to the presence of multiple similar objects that are difficult to distinguish and text prompts that lack detailed descriptions for accurate result determination.

F. Robotic Experiments

Robotic Setup. Our lightweight language-driven grasp detection pipeline is incorporated within a robotic grasping framework that employs a KUKA LBR iiwa R820 robot to deliver quantifiable outcomes. Utilization of the RealSense D435i camera enables the translation of grasping information from LLGD into a 6DoF grasp posture, bearing resemblance

Fig. 3. Visualization of detection results of different language-driven grasp detection methods.

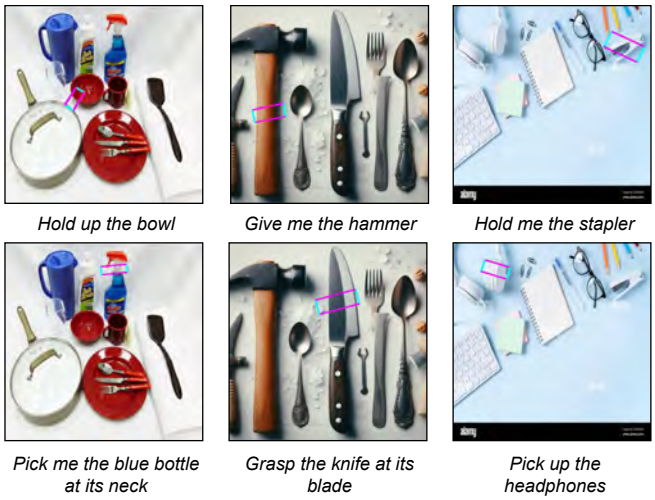

Fig. 4. In the wild detection results. Images are from the internet.

Fig. 5. Prediction failure cases.

to [44]. Subsequently, a trajectory optimization planner [55] is used to execute the grasping action. Experiments were conducted on a table surface for two scenarios: the single object scenario and the cluttered scene scenario, wherein various objects were placed to test each setup. Table III shows the success rate of our method and other baseline models. We can see that our method outperforms other baselines in both single object and cluttered scenarios. Furthermore, our lightweight model allows rapid execution speed without sacrificing the visual grasp detection accuracy.

TABLE III

ROBOTIC LANGUAGE-DRIVEN GRASP DETECTION RESULTS

Baseline	Single	Cluttered
GR-ConvNet [44] + CLIP [13]	0.33	0.30
Det-Seg-Refine [4] + CLIP [13]	0.30	0.23
GG-CNN [54] + CLIP [13]	0.10	0.07
CLIPORT [6]	0.27	0.30
CLIP-Fusion [15]	0.40	0.40
SnapFusion [27]	0.40	0.39
LightGrad [31]	0.41	0.40
LLGD (ours)	**0.43**	**0.42**

V. DISCUSSION

Limitation. Despite achieving notable results in real-time applications, our method still has limitations and predicts incorrect grasp poses in challenging real-world images. Faulty grasp poses are often due to the correlation between the text and the attention map of the visual features not being well-aligned as shown in Fig. 5. From our experiment, we see that when grasp instruction sentences contain rare and challenging nouns that are popular in the dataset, ambiguity in parsing or text prompts would happen and is usually the main cause that leads to the incorrect prediction of grasp poses. Therefore, providing the instruction prompts with clear meanings is essential for the robot to understand and execute the correct grasping action.

Future work. We see several prospects for improvement in future work: *i)* expanding our method to handle 3D space is essential, implementing it for 3D point clouds and RGB-D images to avoid the lack of depth information in robotic applications, *ii)* addressing the gap between the semantic concept of text prompts and input image, analyzing the detailed geometry of objects for distinguishing between items with similar structure, and *iii)* expanding the problem to more complex language-driven manipulation applications, for instance, in case the robots want to grasp a plate containing apples, the robot would need to manipulate the objects in such a manner that prevents the apples from falling.

REFERENCES

[1] Y. Wang, Y. Zheng, B. Gao, and D. Huang, "Double-dot network for antipodal grasp detection," in *IROS*, 2021.

[2] M. Sundermeyer, A. Mousavian, R. Triebel, and D. Fox, "Contact-graspnet: Efficient 6-dof grasp generation in cluttered scenes," in *ICRA*, 2021.

[3] B. Wen, W. Lian, K. Bekris, and S. Schaal, "Catgrasp: Learning category-level task-relevant grasping in clutter from simulation," in *ICRA*, 2022.

[4] S. Ainetter and F. Fraundorfer, "End-to-end trainable deep neural network for robotic grasp detection and semantic segmentation from rgb," in *ICRA*, 2021.

[5] A. Depierre, E. Dellandréa, and L. Chen, "Jacquard: A large scale dataset for robotic grasp detection," in *IROS*, 2018.

[6] M. Shridhar, L. Manuelli, and D. Fox, "Cliport: What and where pathways for robotic manipulation," in *CoRL*, 2022.

[7] F.-J. Chu, R. Xu, and P. A. Vela, "Real-world multiobject, multigrasp detection," *RA-L*, 2018.

[8] T. Nguyen, M. N. Vu, A. Vuong, D. Nguyen, T. Vo, N. Le, and A. Nguyen, "Open-vocabulary affordance detection in 3d point clouds," in *IROS*, 2023.

[9] A. Alliegro, M. Rudorfer, F. Frattin, A. Leonardis, and T. Tommasi, "End-to-end learning to grasp via sampling from object point clouds," *RA-L*, 2022.

[10] Y. Chen, R. Xu, Y. Lin, and P. A. Vela, "A joint network for grasp detection conditioned on natural language commands," in *ICRA*, 2021.

[11] Q. Vuong, S. Levine, H. R. Walke, K. Pertsch, A. Singh, R. Doshi, C. Xu, J. Luo, L. Tan, D. Shah, *et al.*, "Open x-embodiment: Robotic learning datasets and rt-x models," in *CoRL*, 2023.

[12] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, *et al.*, "Improving language understanding by generative pre-training," 2018.

[13] 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 *ICML*, 2021.

[14] G. Tziafas, Y. Xu, A. Goel, M. Kasaei, Z. Li, and H. Kasaei, "Language-guided robot grasping: Clip-based referring grasp synthesis in clutter," *arXiv:2311.05779*, 2023.

[15] 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," *arXiv:2302.12610*, 2023.

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

[17] J. Yang, W. Tan, C. Jin, B. Liu, J. Fu, R. Song, and L. Wang, "Pave the way to grasp anything: Transferring foundation models for universal pick-place robots," *arXiv:2306.05716*, 2023.

[18] A. Brohan, Y. Chebotar, C. Finn, K. Hausman, A. Herzog, D. Ho, J. Ibarz, A. Irpan, E. Jang, R. Julian, *et al.*, "Do as i can, not as i say: Grounding language in robotic affordances," in *CoRL*, 2023.

[19] C. Lynch, A. Wahid, J. Tompson, T. Ding, J. Betker, R. Baruch, T. Armstrong, and P. Florence, "Interactive language: Talking to robots in real time," *RA-L*, 2023.

[20] Y. Lu, Y. Fan, B. Deng, F. Liu, Y. Li, and S. Wang, "Vl-grasp: a 6-dof interactive grasp policy for language-oriented objects in cluttered indoor scenes," in *IROS*, 2023.

[21] H. Ma and D. Huang, "Towards scale balanced 6-dof grasp detection in cluttered scenes," in *CoRL*, 2023.

[22] D. Shah, B. Osiński, S. Levine, *et al.*, "Lm-nav: Robotic navigation with large pre-trained models of language, vision, and action," in *CoRL*, 2023.

[23] H. Ha, P. Florence, and S. Song, "Scaling up and distilling down: Language-guided robot skill acquisition," in *CoRL*, 2023.

[24] S. Chen, P. Sun, Y. Song, and P. Luo, "Diffusiondet: Diffusion model for object detection," in *ICCV*, 2023.

[25] J. Urain, N. Funk, J. Peters, and G. Chalvatzaki, "Se (3)-diffusionfields: Learning smooth cost functions for joint grasp and motion optimization through diffusion," in *ICRA*, 2023.

[26] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *NeurIPS*, 2020.

[27] Y. Li, H. Wang, Q. Jin, J. Hu, P. Chemerys, Y. Fu, Y. Wang, S. Tulyakov, and J. Ren, "Snapfusion: Text-to-image diffusion model on mobile devices within two seconds," *arXiv:2306.00980*, 2023.

[28] T. Salimans and J. Ho, "Progressive distillation for fast sampling of diffusion models," *arXiv:2202.00512*, 2022.

[29] Y. Song, P. Dhariwal, M. Chen, and I. Sutskever, "Consistency models," *arXiv:2303.01469*, 2023.

[30] A. Habibian, A. Ghodrati, N. Fathima, G. Sautiere, R. Garrepalli, F. Porikli, and J. Petersen, "Clockwork diffusion: Efficient generation with model-step distillation," *arXiv:2312.08128*, 2023.

[31] J. Chen, X. Song, Z. Peng, B. Zhang, F. Pan, and Z. Wu, "Lightgrad: Lightweight diffusion probabilistic model for text-to-speech," in *ICASSP*, 2023.

[32] B. Liu, W. Lin, *et al.*, "Rapid diffusion: Building domain-specific text-to-image synthesizers with fast inference speed," in *ACL*, 2023.

[33] A. D. Vuong, M. N. Vu, H. Le, B. Huang, B. Huynh, T. Vo, A. Kugi, and A. Nguyen, "Grasp-anything: Large-scale grasp dataset from foundation models," in *ICRA*, 2023.

[34] Q. Dai, Y. Zhu, Y. Geng, C. Ruan, J. Zhang, and H. Wang, "Graspnerf: Multiview-based 6-dof grasp detection for transparent and specular objects using generalizable nerf," in *ICRA*, 2023.

[35] Y. Chen, Y. Lin, R. Xu, and P. A. Vela, "Keypoint-graspnet: Keypoint-based 6-dof grasp generation from the monocular rgb-d input," in *ICRA*, 2023.

[36] J. Redmon and A. Angelova, "Real-time grasp detection using convolutional neural networks," in *ICRA*, 2015.

[37] S. Kumra and C. Kanan, "Robotic grasp detection using deep convolutional neural networks," in *IROS*, 2017.

[38] P. Ni, W. Zhang, X. Zhu, and Q. Cao, "Pointnet++ grasping: Learning an end-to-end spatial grasp generation algorithm from sparse point clouds," in *ICRA*, 2020.

[39] T. Nguyen, M. N. Vu, B. Huang, A. Vuong, Q. Vuong, N. Le, T. Vo, and A. Nguyen, "Language-driven 6-dof grasp detection using negative prompt guidance," in *ECCV*, 2024.

[40] A. D. Vuong, M. N. Vu, B. Huang, N. Nguyen, H. Le, T. Vo, and A. Nguyen, "Language-driven grasp detection," in *CVPR*, 2024.

[41] V. T. Vo, M. N. Vu, B. Huang, A. Vuong, N. Le, T. Vo, and A. Nguyen, "Language-driven grasp detection with mask-guided attention," in *IROS*, 2024.

[42] 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:2303.05499*, 2023.

[43] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," in *ICLR*, 2021.

[44] S. Kumra, S. Joshi, and F. Sahin, "Antipodal robotic grasping using generative residual convolutional neural network," in *IROS*, 2020.

[45] N. Le, T. Do, K. Do, H. Nguyen, E. Tjiputra, Q. D. Tran, and A. Nguyen, "Controllable group choreography using contrastive diffusion," *TOG*, 2023.

[46] S. Zhang, N. Murray, L. Wang, and P. Koniusz, "Time-reversed diffusion tensor transformer: A new tenet of few-shot object detection," in *ECCV*, 2022.

[47] A. D. Vuong, M. N. Vu, T. Nguyen, B. Huang, D. Nguyen, T. Vo, and A. Nguyen, "Language-driven scene synthesis using multi-conditional diffusion model," in *NeurIPS*, 2024.

[48] J. Li, R. Selvaraju, A. Gotmare, S. Joty, C. Xiong, and S. C. H. Hoi, "Align before fuse: Vision and language representation learning with momentum distillation," *NeurIPS*, 2021.

[49] P. Vincent, "A connection between score matching and denoising autoencoders," *Neural computation*, 2011.

[50] G. K. Gupta, R. Sacks-Davis, and P. E. Tescher, "A review of recent developments in solving odes," *CSUR*, 1985.

[51] 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:2010.11929*, 2020.

[52] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv:1810.04805*, 2018.

[53] K. Zhou, J. Yang, C. C. Loy, and Z. Liu, "Conditional prompt learning for vision-language models," in *CVPR*, 2022.

[54] D. Morrison, P. Corke, and J. Leitner, "Closing the loop for robotic grasping: A real-time, generative grasp synthesis approach," *arXiv:1804.05172*, 2018.

[55] M. N. Vu, F. Beck, M. Schwegel, C. Hartl-Nesic, A. Nguyen, and A. Kugi, "Machine learning-based framework for optimally solving the analytical inverse kinematics for redundant manipulators," *Mechatronics*, 2023.