# DM-VLP-Grasp: Diffusion Model-Based Grasp Planning with Visual-Language Pretraining for Unknown Object Manipulation

Changyu Li[1], Gao Liu[1], Ruchao Liao[1], Linkun Yu[2], Jianguo Zhang[2], Ning Ding[2*]
[1]Guangdong Power Grid Corporation, Guangzhou Guangdong, 51000, China
[2]The Chinese University of Hong Kong, Shenzhen Guangdong, 518000, China
E-mail: changyuligsc@163.com
[*]Corresponding author

**Keywords:** diffusion model, visual language pre-training, unknown object grasping, grasping strategy generation, robot vision

**Received:** April 25, 2025

*This paper proposes an unknown object grasping algorithm (DM-VLP-Grasp) based on diffusion model and visual language pre-training, aiming to improve the grasping performance of robots in complex environments. By improving the visual language pre-training model, the image and text information are integrated to accurately extract the object grasping features; the diffusion model is used to generate a reliable grasping strategy, and efficient grasping is achieved through iterative optimization. On a self-built dataset containing 8000 samples, the results show that the grasping success rate of DM-VLP-Grasp reaches 93.6%, and the single strategy generation time is 0.78 seconds, showing high stability and computational efficiency. The grasping stability is measured by the root mean square value (RMS) of the object shaking amplitude and the grasping force fluctuation range, both of which show excellent performance. The experimental results verify the effectiveness and innovation of the algorithm in the unknown object grasping task, and provide a new solution for robot automated grasping.*

*Povzetek: Opisan jei algoritem DM-VLP-Grasp za prijemanje neznanih objektov, ki združuje difuzijski model in vizualno-jezikovno predusposabljanje za stabilno, semantično vodeno robotsko manipulacijo.*

## 1 Introduction

Robot automatic grasping technology is increasingly used in industrial production, logistics, warehousing, and home services. In industrial scenarios, robots need to quickly grasp parts of various shapes and sizes to complete assembly tasks; in the process of logistics warehousing, facing a large number of goods with different packaging, effective grasping is the prerequisite for automatic sorting; in home service scenarios, the grasping ability of unknown daily items directly affects the service quality of the robot. Among them, reliable grasping of unknown objects is the key to the intelligence and autonomy of robots, and it is also the core of robots' adaptation to complex environments.

Existing unknown object grasping technologies have certain limitations. Model-based grasping methods represented by geometric models achieve grasping point and pose planning by establishing a three-dimensional geometric model of the target. Still, their accuracy and efficiency will be significantly affected when encountering targets with irregular and complex shapes [1]. For example, for targets with features such as holes and concave-convex textures, point cloud-based geometric reconstruction algorithms often have missing models or errors, making subsequent grasping planning impossible [2]. The grasping planning method is based on the physical model and considers the physical properties of the target (mass, center of gravity, friction, etc.). Still, there are problems, such as difficulty in accurately obtaining the target's physical parameters and adapting to the dynamic changes of the target properties quickly. For example, the friction coefficient is complex to update in real time with the traditional physical model due to the influence of factors such as humidity and stains on the object's surface.

Although the grasping method based on deep learning can improve the grasping performance, such as the grasping algorithm based on convolutional neural network needs to learn the mapping relationship between the target visual features and grasping strategy from the training samples, it is too dependent on the training data, resulting in insufficient generalization ability for unknown targets [3]. Studies have shown that when there are significant differences between the test object and the training sample in terms of shape, texture, etc., the success rate of the grasping algorithm based on CNN will be reduced by 30%.

The grasping strategy based on reinforcement learning is optimized interactively with the environment, but there are problems such as long training time and poor convergence stability [4]. In complex environments, reinforcement learning algorithms often go through millions of iterations to find the optimization strategy. They are prone to fall into local extremes, making it difficult to cope with the challenges of target diversity and environmental changes in real scenes. These limitations make it difficult for existing technologies to cope with challenges in complex and changing shapes and materials.

Scholars at home and abroad have done a lot of research on the grasping problem of unknown targets. In traditional methods, the target geometric model is constructed based on point cloud data and three-dimensional grids, and the candidate points are grasped in combination with heuristic rules [5]. There is a problem that modeling errors can easily cause grasping failures. For example, the point cloud-based target recognition and grasping planning method proposed by Rusu et al. can handle geometric targets well. Still, its grasping success rate is significantly reduced for targets with complex topological structures [6]. The physical model method optimizes the grasping strategy by simulating the force state of the object, but the error in parameter estimation will affect the reliability of the grasping strategy. For example, the work done by Mousavian et al. shows that the grasping planning method based on physical simulation can produce reasonable grasping strategies under ideal parameter settings. Still, in actual operation, the grasping planning failure rate is as high as 40% due to parameter estimation deviations. In deep learning, convolutional neural networks have achieved good results in grasping detection with their powerful feature extraction capabilities [7]. However, existing methods mainly rely on visual information and lack understanding of the semantics and physical properties of the target. Reinforcement learning is driven by reward mechanisms, represented by deep QNN and its variants, to achieve efficient grasping learning in a simulation environment [8]. However, Lillicrap et al.'s research shows a domain difference between simulated and real scenes in the grasping algorithm based on reinforcement learning, resulting in a more than 50% performance drop. Deep learning methods are still insufficient in processing complex scenes and semantic information [9].

This paper proposes an unknown object grasping algorithm based on diffusion model and visual language pre-training (DM-VLP-Grasp), aiming to improve the grasping performance of robots in complex environments. The core question of this study is: "Can the integration of visual language pre-training (VLP) improve the semantic generalization ability of unknown object grasping?" By combining visual language pre-training with diffusion model, we expect to significantly improve the grasping success rate and stability of robots for unknown objects, while improving the computational efficiency of the algorithm. Specifically, the main objectives of this paper include:

1) Propose an unknown object grasping algorithm framework to enhance the semantic understanding of unknown objects by fusing multimodal information (images and text).

2) Verify the efficiency and robustness of the algorithm on a self-built dataset to ensure that it can perform well on objects of different shapes and materials.

3) Through ablation studies, prove the effectiveness of multimodal fusion strategy in improving grasping performance.

Compared with traditional methods, DM-VLP-Grasp has significant advantages in grasping success rate, stability and computational efficiency, especially showing higher adaptability and stability when dealing with irregular or deformable objects.

## 2 Related theoretical and technical foundations

In the field of robotic grasping, a variety of methods have been proposed to cope with grasping tasks in different scenarios. Table 1 summarizes the current state-of-the-art (SOTA) grasping techniques, including geometry-based methods, physical model-based methods, convolutional neural network (CNN)-based methods, reinforcement learning methods, and diffusion model-based methods. These methods have their own advantages and disadvantages in terms of success rate, adaptability to unknown objects, computational efficiency, and data dependence.

Table 1: Performance comparison of different crawling methods

| Method type | Success rate | Adaptability | Computational efficiency | Data Dependency |
|---|---|---|---|---|
| **Geometric model** | 61.2% | Low | High | Low |
| **Physical model** | 68.5% | Medium | Medium | Medium |
| **CNN** | 81.3% | Medium | High | High |
| **Reinforcement learning** | 84.7% | High | Low | High |
| **Diffusion model** | 93.6% | High | High | Medium |

Geometric model: Plan the grasping points and postures by building a 3D geometric model of the target.

This method works well when dealing with objects with regular geometric shapes, but its accuracy and efficiency will drop significantly for objects with irregular and complex shapes.

Physical model: Consider the physical properties of the target (such as mass, center of gravity, friction coefficient, etc.) to optimize the grasping strategy. However, it is difficult to accurately obtain the physical parameters of the target and it is difficult to adapt to the dynamic changes of the target properties.

CNN method: Use convolutional neural networks to learn grasping strategies from visual features. Although it performs well in feature extraction, it has a strong dependence on training data, resulting in insufficient generalization ability for unknown targets.

Reinforcement learning method: Optimize grasping strategies through interaction with the environment. Although it has achieved good results in simulated environments, it has a long training time, poor convergence stability, and obvious performance degradation in real scenes.

Diffusion model: Generate grasping strategies by gradually adding noise and inverse denoising. This method performs well in generating high-quality grasping strategies and has strong generalization ability for unknown objects.

Compared with the existing technology, DM-VLP-Grasp fills the gap in generalization ability and multimodal information fusion by combining diffusion model and visual language pre-training. Diffusion model can generate high-quality grasping strategies, while visual language pre-training enhances the semantic understanding of unknown objects.

## 2.1 Principles and applications of the diffusion model

### 2.1.1 Basic concepts of the diffusion model

Inspired by the abstraction and extension of diffusion phenomena in the physical world, the core idea of diffusion model research is to learn the distribution law within the data by gradually adding noise and inversion denoising [10]. In the positive diffusion stage, the model takes the original data as the initial state, just like dropping a drop of ink into clear water [11]. The data will gradually lose its original characteristics as time passes and eventually evolve into a completely random noise distribution. This process has a certain degree of predictability and adopts a reasonable noise addition strategy to control data changes precisely.

Inversion denoising is like gradually purifying turbid water. In the case of full noise, the powerful fitting ability of neural networks is used to progressively eliminate noise and restore the original data. In this process, the neural network must constantly learn and judge how much noise must be removed at each step to be as close to the real data as possible. This ability to reconstruct data from noise gives it a unique advantage in data generation.

### 2.1.2 Application of the diffusion model in computer vision

In the study of computer vision, diffusion models have achieved good results in many fields. For example, Stable Diffusion can automatically generate images with high realism and semantic consistency based on the text description entered by the user, achieving accurate reproduction from fantasy to real scenes. In terms of image repair, the algorithm can reasonably infer and fill in the missing parts based on the residual information of the damaged image, thereby achieving image integrity [12]. In super-resolution tasks, the diffusion model can increase the details of low-resolution images and improve the clarity of the image.

The research results of this project will provide a theoretical basis for applying scattering models in target modeling, posture estimation, and other fields. For the vast search space composed of the combination of target posture and morphology in the unknown target grasping task, the progressive generation characteristics of the diffusion model can be used to systematically explore the space, generate multiple possible target posture hypotheses, provide rich alternatives for the grasping strategy, and help the robot find the best grasping method.

## 2.2 Overview of visual studio pre-training technology

### 2.2.1 Basic framework of pre-training

The visual language pre-training model is an essential tool that integrates visual and linguistic information. The visual feature extraction module mainly converts images into feature expressions that computers can understand. Convolutional neural networks (CNNs) and visual transformers (ViTs) are widely used. CNN uses convolution and pooling methods to extract local and overall features of images gradually; ViT divides images into several small blocks and uses the Transformer framework to obtain the connection between each component.

Among them, the language feature extraction module with the Transformer encoder as the core can realize deep semantic analysis of the input text and understand its grammatical structure and semantic meaning. Cross-modal fusion is a bridge connecting visual and language features [13]. The cross-modal fusion method based on the attention mechanism can enable the model to focus on the relevant parts of the image and text, thereby realizing effective interaction of cross-modal information. For example, the CLIP model uses a contrastive learning method to align the image and text representations in the same feature space, so that the model can understand the

correspondence between the image and the text.

#### 2.2.2 Application of visual language pre-training in the field of robotics

In robotics, visual language pre-training is a significant research direction. When navigating, the robot can plan a reasonable path based on natural language instructions and environmental images [14]. For example, when the user issues a command "bypass the desktop and go to the window", the robot will understand the semantics of the command based on the pre-trained visual language, and then combine the real-time image of the surrounding environment to avoid obstacles and achieve navigation accurately.

In target recognition, when the target is difficult to identify based on visual features alone, the model can use relevant text descriptions, such as color, purpose, etc., to assist in completing the recognition task. This project proposes a robot grasping model based on visual information for robot grasping tasks. The model can effectively integrate the physical properties of objects (weight, hardness, etc.) and grasping features (such as grasping position, grasping strength, etc.), thereby helping the robot to judge the grasping target accurately and providing new ideas for further improving the grasping model.

### 2.3 Key technologies for grasping unknown objects

#### 2.3.1 Object feature extraction technology

In the process of extracting unknown targets, the extraction of target features is a key step. The traditional edge detection method finds pixel boundaries with apparent changes in the image [15]. Its typical representative is the Canny edge detection algorithm, which can extract the target boundary under a simple background. Still, it is easy to have edge breakage or false detection under the influence of factors such as lighting and texture. Shape descriptors use mathematical methods to quantify the shape of the target, but the description effect of irregular and dynamically changing targets is not ideal.

In recent years, feature extraction methods based on deep learning have developed rapidly. Point cloud feature extraction technology can directly process point cloud data. For example, PointNet can effectively learn the overall features of point clouds. Still, point cloud data has problems such as sparsity and noise, leading to decreased feature extraction accuracy. Image-based feature extraction methods entirely use the powerful feature learning ability of convolutional neural networks and can extract the visual features of targets from two-dimensional images [16]. However, the lack of depth information makes it difficult to accurately judge the target's spatial position, and the target's physical attribute information cannot be directly obtained.

#### 2.3.2 Generation of grasping strategies

The existing grasping strategy generation methods can be divided into rule-based grasping strategies and learning-based grasping strategies. Rule-based grasping rules are formulated according to the geometric shape and size of the object. This method is simple, intuitive, and has a fast calculation speed, but its application scope is limited to objects with simple shapes and known types.

Among the learning-based methods, reinforcement learning is a method that gradually optimizes the grasping strategy by constantly trying to grasp the rewards or penalties in the environment. Still, it requires a large amount of training data, has a large amount of calculation, a long training cycle, and is prone to falling into local extremes [17]. Imitation learning is a strategy for obtaining grasping actions from human demonstrations. Still, in practical applications, there are problems such as a large amount of high-quality labeled data that is difficult to obtain and an insufficient generalization ability of the model for targets significantly different from the training samples.

Given the shortcomings of existing methods in dealing with unknown targets, a grasping strategy generation idea based on a diffusion model is proposed. This project intends to use the powerful generalization ability of the diffusion model, combined with the rich semantic and visual information provided by the visual language pre-training model, to explore more efficient grasping strategies and offer new ideas for grasping complex targets in complex environments [18].

## 3 Design of an unknown target grasping algorithm based on the diffusion model and visual language pre-training

### 3.1 Algorithm architecture

In the process of grasping strategy generation, the diffusion model gradually adds noise and reversely denoises, which is similar to gradually optimizing the solution in a noisy environment. This process not only improves the quality of the generated strategy, but also ensures the feasibility and stability of the strategy [19].

The visual-language pretraining (VLP) module processes images and text to output $F_{\text{vlp}}$ , a fused multimodal feature. A grasp-task-specific layer then refines $F_{\text{vlp}}$ into $F_{\text{vlp}}^{*}$ (Equation 4), which undergoes feature optimization (self-supervised contrastive learning, Equation 5) and dimensionality reduction (PCA + linear transformation, Equation 6) to produce $F_{\text{grasp}}$ . The diffusion model takes $F_{\text{grasp}}$ as input to generate initial strategies $S_{\text{init}}$ , which are further optimized by the genetic algorithm to produce $S_{\text{opt}}$ . This clarifies that:

$F_{\text{vlp}}$ : Raw multimodal fusion output

$F_{\text{vlp}}^{*}$ : Task-refined features after VLP-specific layer

$F_{\text{grasp}}$ : Optimized, low-dimensional features for diffusion

## 3.2 Object feature extraction module based on visual language pre-training

### 3.2.1 Improvement of pre-training model integrating multimodal information

This paper makes innovative improvements to the existing model, introduces the multi-head self-attention mechanism (MHSA) and a specific layer for grasping tasks, and builds a pre-trained model architecture that is more suitable for grasping tasks. With these improvements, the model can more effectively integrate multimodal information and extract features directly related to grasping tasks.

In the improved model, when processing image features **I** and text features **T**, features are first extracted through independent encoders $Enc_I$ and $Enc_T$ respectively. $Enc_I$ can use the classic convolutional neural network architecture, such as the ResNet series, to gradually extract local and global features of the image through multi-layer convolution and pooling operations to obtain $\mathbf{f}_I$; $Enc_T$ is based on the Transformer encoder, which performs deep semantic analysis on the text, captures the grammatical structure and semantic connotation in the text, and generates $\mathbf{f}_T$.

$$\begin{aligned}\mathbf{f}_I &= Enc_I(\mathbf{I})\\ \mathbf{f}_T &= Enc_T(\mathbf{T})\end{aligned} \tag{1}$$

Subsequently, the cross-modal attention weight $\mathbf{W}_{vl}$ is calculated using the multi-head self-attention mechanism. The multi-head self-attention mechanism can capture the association between image and text features from multiple angles, and has stronger expressive power than the traditional attention mechanism. During the calculation process, the attention weight matrix $\mathbf{W}_{vl}$ is obtained by performing matrix operations on $\mathbf{f}_I$ and$\mathbf{f}_T$, dividing them by $\sqrt{d_k}$ for scale scaling, and then normalizing them through the softmax function. Each element in the matrix represents the degree of association between image features and text features. The larger the value, the stronger the correlation between the two.

$$\mathbf{W}_{vl} = \text{softmax}\left(\frac{\boldsymbol{f}_I \cdot \boldsymbol{f}_T^T}{\sqrt{d_k}}\right) \tag{2}$$

Among them, $d_k$ is the feature dimension, which is set to balance the computational complexity and model performance, and avoid problems such as excessive computation or gradient disappearance due to high dimensions.

The image features$\mathbf{f}_I$ are weighted and summed, and added to the text features $\mathbf{f}_T$to achieve deep fusion of multimodal information and obtain $\mathbf{F}_{vlp}$. This process enables the model to utilize the complementarity between image and text information fully. For example, when some details of an object in an image are unclear due to occlusion, the text description can provide additional clues to help the model understand the object more accurately.

$$\mathbf{F}_{vlp} = \mathbf{W}_{vl} \cdot \mathbf{f}_I + \mathbf{f}_T \tag{3}$$

A grasping-task-specific layer is added at the end of the model to enhance the model's sensitivity to grasping-related features. This layer contains a set of learnable parameters $\theta_{\text{grasp}}$, which adjust the features by matrix multiplication with $\mathbf{F}_{vlp}$ and introduce nonlinearity using the ReLU activation function to obtain $\mathbf{F}_{vlp}^{*}$. The learnable parameters $\theta_{\text{grasp}}$ can be automatically optimized according to the needs of the grasping task during model training, allowing the model to focus more on extracting features that have an essential impact on grasping decisions, such as stable grasping areas and weak points of objects.

$$\mathbf{F}_{vlp}^{*} = \text{ReLU}\left(\boldsymbol{F}_{vlp} \cdot \theta_{grasp}\right) \tag{4}$$

### 3.2.2 Feature optimization for grasping tasks

After obtaining the feature $\mathbf{F}_{vlp}^{*}$ that integrates multimodal information, although it already contains rich semantic and visual information, these features are still redundant. They may not fully adapt to the needs of grasping strategy generation. Therefore, it is necessary to further process them by designing feature optimization algorithms to improve the quality and effectiveness of features.

This paper adopts a self-supervised contrastive learning method to explore the potential relationship between features [20]. The core idea of self-supervised contrastive learning is to construct positive and negative sample pairs so that the model can distinguish similar samples from dissimilar samples, automatically discovering the data's intrinsic structure and feature association. The contrast loss function $\mathcal{L}_{\text{contrast}}$ is defined, and its calculation process is based on the mutual information principle in information theory, which aims to maximize the similarity between different feature representations of the same object, while minimizing the similarity between feature representations of other objects.

$$\mathcal{L}_{\text{contrast}} = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{exp\left(\frac{sim\left(\boldsymbol{F}_{vlp}^{i}, \boldsymbol{F}_{vlp}^{i+}\right)}{\tau}\right)}{\sum_{j=1}^{2N} exp\left(\frac{sim\left(\boldsymbol{F}_{vlp}^{i}, \boldsymbol{F}_{vlp}^{j}\right)}{\tau}\right)} \tag{5}$$

Among them, $N$ is the number of samples, $\mathbf{F}_{vlp}^{i+}$ is the positive sample feature belonging to the same object as $\mathbf{F}_{vlp}^{i}$, $\mathbf{F}_{vlp}^{j}$ is the negative sample feature, sim is the cosine similarity function, which is used to measure the angle between two feature vectors [21]. The smaller the angle, the more similar the features are; $\tau$ is the temperature parameter, which is used to adjust the steepness of the contrast loss function and control the difficulty of model learning. A smaller $\tau$ value will make the model more strictly distinguish between positive and negative samples.

At the same time, to reduce the feature dimension and improve the computational efficiency of subsequent grasping strategy generation, the high-dimensional feature $\mathbf{F}_{vlp}^{*}$ is converted into a low-dimensional representation $\mathbf{F}_{\text{grasp}}$ through the feature mapping function $M(\cdot)$. The mapping function $M(\cdot)$ is constructed based on principal component analysis (PCA) and linear transformation. PCA is a commonly used dimensionality reduction method. It decomposes the feature covariance matrix, finds the main component directions in the data, and projects the original high-dimensional data onto these main component directions, thereby reducing dimensionality while retaining most of the information. Linear transformation further adjusts the features after PCA processing to make them more in line with the requirements of the grasping task, such as highlighting the feature dimensions related to grasping stability.

$$\mathbf{F}_{\text{grasp}} = M\left(\mathbf{F}_{vlp}^{*}\right) \tag{6}$$

## 3.3 Grasping strategy generation module based on diffusion model

### 3.3.1 Design of grasping strategy diffusion generation mechanism

The constraint function Constraint $(S)$ is applied as a post-processing filter after the diffusion model generates $S_{\text{init}}$. It ensures:

- Grasp points lie within object convex hull (geometric constraint)
- Grasp force $f$ is within the gripper's safe range ( $0.1 - 5.0$ N )
- Contact normals are aligned with gripper jaws (orientation constraint)

Strategies violating these constraints are rejected, and the top 50 valid $S_{\text{init}}$ are fed into the genetic algorithm. This separation of diffusion-based generation and constraint-based filtering maintains the model's generative flexibility while ensuring physical feasibility.

The objective function is designed to generate high-quality grasping strategies by gradually adding noise and reversing the denoising process. Specifically, the objective function of the diffusion model can be expressed as:

$$\mathcal{L}_{\text{diffusion}} = \mathbb{E}_{t,\mathbf{x}_0}[\|\mathbf{x}_t - \mathbf{x}_{t-1}\|^2] \tag{7}$$

where $\mathbf{x}_t$ represents the state at time step $t$, and $\mathbf{x}_{t-1}$ represents the state at time step $t-1$. This objective function ensures that the generated grasping strategy gradually approaches the true data distribution during the denoising process by minimizing the difference between adjacent time steps. This process not only improves the quality of the generated strategy, but also ensures the feasibility and stability of the strategy.

In the forward diffusion process, the initial strategy $\mathbf{x}_0$ is gradually covered by the noise $\epsilon_t$, and eventually evolves into a completely random noise distribution:

$$\mathbf{x}_t = \sqrt{1-\beta_t}\mathbf{x}_{t-1} + \sqrt{\beta_t}\epsilon_t \tag{8}$$

where $\beta_t$ is the diffusion coefficient, which controls the amount of noise added at each step. In the reverse denoising process, the model gradually recovers the initial strategy $\mathbf{x}_0$ by estimating the noise $\epsilon_t$:

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{1-\beta_t}}\left(\mathbf{x}_t - \sqrt{\beta_t}\epsilon_t\right) \tag{9}$$

Through this step-by-step denoising process, the model is able to generate high-quality grasping strategies. The diffusion model takes the grasp-relevant feature representation $F_{\text{grasp}}$ as input and generates grasp strategies in a continuous 6 -dimensional space (3D coordinates $p = (x, y, z)$, 3D angles $\theta = \left(\theta_x, \theta_y, \theta_z\right)$, and grasp force $f$ ). The input distribution is conditioned on object features, while the output distribution approximates the real grasp strategy distribution. During training, the model minimizes the mean squared error (MSE) loss between predicted noise and actual noise added at each diffusion step:

$$L_{\text{diffusion}} = \mathbb{E}_{t,x_0,\epsilon\sim\mathcal{N}(0,1)}[\|\epsilon - \epsilon_\theta(x_t, t)\|_2^2] \tag{10}$$

where $x_t$ is the noisy strategy at step $t$, $\epsilon_\theta$ is the noise prediction network, and $t$ is the diffusion time step. This iterative denoising process gradually refines random noise into plausible grasp strategies, guided by the learned feature-conditioned distribution.

#### 3.3.2 Optimization and screening of generation strategies

The genetic algorithm initializes its population with 50 strategies generated by the diffusion model ( $S_{\text{init}}$ ), expanded to form a diverse pool. Each generation evolves this population via crossover and mutation, with fitness evaluated by $F(S)$ (Equation 10). The final $S_{\text{opt}}$ is t-fitness strategy from the last generation's population, ensuring the diffusion models output serves as both the initial solution and the foundation for population-based optimization.

In the grasping strategy optimization process, the objective function is designed to further improve the performance of the generated strategy. Specifically, the optimization objective function can be expressed as:

$$\mathcal{L}_{\text{optimization}} = \alpha \cdot \text{SuccessRate} + \beta \cdot \text{Stability} + \gamma \cdot \text{Efficiency} \quad (10)$$

Where SuccessRate represents the grasping success rate, Stability represents the grasping stability, and Efficiency represents the computational efficiency. The weights α, β, and γ are used to balance the importance of different indicators and are adjusted according to specific task requirements. For example, when handling fragile objects, the weight of β can be increased to ensure the stability of the grasping process; in scenarios that require fast response, the weight of γ can be increased to improve computational efficiency.

In the unknown object grasping task, the goal is to improve the grasping success rate, stability, and computational efficiency. Therefore, the design of the diffusion model and the optimization objective function closely revolves around these goals. The diffusion model generates high-quality grasping strategies through step-by-step denoising to ensure the feasibility and stability of the strategy in practical applications. The optimization objective function further adjusts the strategy to meet the needs of specific tasks, such as balancing the success rate, stability, and computational efficiency by adjusting the weights when handling objects of different shapes and materials.

The genetic algorithm (GA) optimization employs the following hyperparameters:

- Population size: 50 individuals per generation
- Mutation rate: 5% (Gaussian mutation with $\sigma = 0.1$ for continuous parameters)
- Crossover rate: 80% (arithmetic crossover for real-valued vectors)
- Generations: 20 iterations

The initial population is seeded with 50 strategies generated by the diffusion model ( $S_{\text{init}}$ ), expanded to form a diverse set of candidate solutions. The fitness function $F(S)$ (Equation 10) balances success rate ( $\alpha = 0.5$ ), stability ( $\beta = 0.3$ ), and efficiency ( $\gamma = 0.2$ ), with weights tuned via cross-validation. Strategies are selected using tournament selection ( $k = 3$ ), and the top 10% of each generation are elitist-preserved to the next iteration.

# 4 Experimental design and simulation

## 4.1 Experimental environment and data set

#### 4.1.1 Hardware environment and software tools

This experiment was conducted on a high-performance workstation with the following hardware configurations: Intel Core i9 - 13900K processor with 24 cores and 32 threads, which can meet the multi-threaded parallel computing requirements of complex algorithms; 64GB DDR5 memory to ensure data storage and fast reading; NVIDIA RTX 4090 graphics card with 16384 CUDA cores and 24GB GDDR6X video memory, providing powerful parallel computing capabilities for deep learning model training and reasoning.

Regarding software tools, PyTorch 2.0 is used as the deep learning framework. Its dynamic computational graph feature facilitates algorithm models' rapid iteration and debugging. It is combined with CUDA 12.1 and cuDNN 8.9 acceleration libraries to give full play to GPU computing performance. Gazebo 11 is used as the simulation platform, which can build a highly realistic physical simulation environment, accurately simulate the interaction between robots, objects and the environment, and support the setting of physical properties of objects of different materials and shapes, such as friction coefficient, mass distribution, etc., to provide a close-to-real test scenario for the grasping experiment [22]. The data processing tool uses Open3D in the Python ecosystem to process point cloud data, which can realize operations such as point cloud filtering, registration, and surface reconstruction; OpenCV is used for image preprocessing, including image enhancement, edge detection, etc., to improve data quality to meet the algorithm input requirements.

#### 4.1.2 Introduction to the self-built unknown object grasping data set

The self-built dataset contains **8,000 grasp configurations** for 1,200 unique objects (12 categories, 50–100 objects per category), with each object annotated with 5–8 feasible grasp points across different orientations. Each "sample" represents a single grasp configuration (image, point cloud, and annotated strategy). The dataset is split into an 80% training set, 10% validation set, and 10% test set, with stratified sampling to ensure category balance. Cross-validation (5-fold) was used during hyperparameter tuning.

Data annotation follows strict specifications, and professionals combine the object's geometric structure and physical characteristics to annotate the object grasping point, grasping direction, object category and other information. The grasping point annotation is based on the stable grasping area of the object to ensure that the annotation point is feasible in actual grasping; the grasping direction annotation is accurate to 5° intervals, covering the 360° spatial range of the object; the object categories cover 12 categories, including metal products, plastic products, glassware, fabrics, etc., each category contains 50-100 samples to ensure data diversity. The dataset is used as a core test resource in the experiment for algorithm training, verification, and comparative testing. Its rich samples and fine annotations can effectively test the algorithm's adaptability to different types of unknown objects.

## 4.2 Comparative experiment settings

### 4.2.1 Comparison with Traditional Grasping Algorithms

The grasping algorithm based on geometric models (GM-Grasp) and the grasping algorithm based on physical models (PM-Grasp) are selected as the traditional algorithm comparison objects. The GM-Grasp algorithm extracts the convex hull and concave points of the object's three-dimensional point cloud and generates grasping candidates in combination with heuristic rules. In the experiment, the convex hull decomposition accuracy is set to 0.1mm, and the concave point detection threshold is set to 0.05mm; the PM-Grasp algorithm uses the object's mass, center of gravity, and friction coefficient to simulate the force condition, and uses the optimization algorithm to find a stable grasping posture. The friction coefficient value range in the parameter setting is 0.1-0.8, and the step length is 0.05.

These two algorithms are selected for comparison because they represent the mainstream technical route of traditional grasping methods. In contrast, the advantages of the algorithm in this paper can be verified in dealing with objects with complex shapes and unknown physical properties, focusing on evaluating the grasping success rate and stability indicators, and analyzing the limitations of traditional methods when facing diverse objects.

### 4.2.2 Comparison with advanced grasping algorithms based on deep learning

The grasping algorithm based on a convolutional neural network (CNN-Grasp) and the grasping algorithm based on reinforcement learning (RL-Grasp) are selected for comparison. CNN-Grasp uses the improved ResNet50 as the backbone network, outputs the position, angle and quality score of the grasping box, and sets the learning rate to 0.001 and the batch size to 32 during training; RL-Grasp is based on the deep Q network (DQN), uses the success or failure of grasping as the reward signal, explores the environment for 1 million steps for training, and the discount factor is 0.99.

Compared with these advanced deep learning algorithms, the purpose is to verify the innovative advantages of this algorithm in integrating multimodal information and the strategy generation mechanism. The comparative experiment is carried out from multiple dimensions, such as grasping success rate and computational efficiency, to evaluate the comprehensive performance of this algorithm in complex scenarios.

## 4.3 Evaluation indicators and methods

### 4.3.1 Grasping success rate

The grasping success rate is the ratio of successful grasps to the total number of grasps. In the experiment, a double standard is used to judge whether a grasping is successful: first, the robot needs to complete the grasping action within 3 seconds, and the object must not fall within 5 seconds after grasping; second, the grasping force is monitored in real time through the force sensor. If the grasping force is within the preset safety threshold (dynamically calculated according to the object's mass), the grasping is considered successful. The evaluation method combines the automatic detection program with manual review. The automatic detection program makes a preliminary judgment based on the sensor data and preset rules, and the manual review ensures the accuracy of the results and reduces misjudgment.

### 4.3.2 Grasping stability

Grasping stability is measured by two indicators: the object shaking amplitude and the grasping force fluctuation range. The object shaking amplitude uses the inertial measurement unit (IMU) installed at the end of the robot to collect data and calculate the root mean square value of the angular velocity and angular acceleration of the object in the X, Y, and Z axes; the grasping force fluctuation range records the maximum and minimum force values of the force sensor during the grasping process, and calculates the ratio of the difference to the average force value. Through the quantitative analysis of these two indicators, the stability of the grasping strategy during execution can be intuitively evaluated. The smaller the indicator value, the higher the stability.

### 4.3.3 Computational efficiency

The single grasping strategy generation time and memory usage during algorithm operation evaluate the computational efficiency. The single grasping strategy generation time is obtained by recording the time interval from the input data to the output of the final grasping strategy, accurate to milliseconds; the memory usage uses system performance monitoring tools (such as NVIDIA's nvvp tool) to monitor the GPU video memory and system memory usage in real time. In the

comparative experiment, the algorithm's real-time performance and resource consumption in practical applications are evaluated by comparing these two indicators of different algorithms.

## 4.4 Experimental results and analysis

### 4.4.1 Quantitative analysis

**Model architecture details**:

- **Visual encoder**: ResNet-50 with feature pyramid network (FPN), pretrained on ImageNet, outputting 1,024-dimensional visual features.
- **Language encoder**: 6-layer Transformer encoder with 512-dimensional hidden states, pretrained on ConceptNet.
- **Diffusion model**: U-Net architecture with 4 downsampling/upsampling blocks, attention layers at bottleneck, trained with 1,000 diffusion steps.

**Training parameters**:

- **Optimizer**: AdamW with weight decay 1e-4
- **Learning rate**: 1e-4 (warmup for 10 epochs)
- **Batch size**: 64
- **Epochs**: 100
- **Training time**: ~48 hours on NVIDIA RTX 4090

To evaluate the stability and generalization ability of the algorithm, we used multiple random seeds (10 seeds in total) for experiments. Each seed corresponds to a different set of random initialization and data partitioning to ensure the diversity of results. The dataset contains 12 different categories of objects, each category contains 50-100 samples. In each category, we randomly selected 5 objects for testing to ensure that each category has enough samples. For each object in each category, we conducted 20 repeated experiments to evaluate the performance of the algorithm under different conditions. We conducted a total of 2,000 experiments (12 categories × 5 objects/category × 20 repeated experiments/object = 1,200 experiments). The standard deviation is calculated by calculating the variance of the results of these 2,000 experiments. To further verify the statistical significance of the results, we performed independent sample t-tests on all key indicators. Specific indicators include:

• Grasp success rate: the ratio of successful grasps to the total grasps.

• Grasp stability: measured by the root mean square value (RMS) of the object shaking amplitude and the grasping force fluctuation range.

• Strategy generation time: the time from inputting data to outputting the final grasping strategy.

The results of the independent sample t-test show that the differences between the DM-VLP-Grasp algorithm and other comparison algorithms are statistically significant in all indicators ($p < 0.01$). This indicates that our algorithm is significantly better than existing methods in terms of grasping success rate, stability, and computational efficiency. Table 2 shows that the DM-VLP-Grasp algorithm performs well in all key indicators with low standard deviations, indicating that the results are highly stable and reliable.

Table 2: Experimental results statistics

| Metrics | DM-VLP-Grasp | CNN-Grasp | RL-Grasp | GM-Grasp | PM-Grasp |
|---|---|---|---|---|---|
| **Success rate (%)** | 93.6 ± 2.1 | 81.3 ± 2.7 | 84.7 ± 2.4 | 61.2 ± 3.5 | 68.5 ± 3.2 |
| **Stability (RMS swing)** | 0.08 ± 0.01 | 0.15 ± 0.02 | 0.13 ± 0.01 | 0.25 ± 0.03 | 0.21 ± 0.02 |
| **Strategy generation time (s)** | 0.78 ± 0.05 | 1.1 ± 0.08 | 3.2 ± 0.3 | 1.5 ± 0.1 | 1.8 ± 0.2 |

Figure 1 shows the trend of the success rate of different algorithms when handling objects of different shapes (sphere, cube, irregular shape). As can be seen from the figure, the DM-VLP-Grasp algorithm maintains a high success rate on objects of all shapes, especially on irregular objects, with a success rate 12.4% higher than the second-best algorithm, RL-Grasp. In contrast, traditional algorithms such as GM-Grasp and PM-Grasp have lower success rates on irregular objects, which shows that the DM-VLP-Grasp algorithm has stronger adaptability and robustness when handling objects of complex shapes.

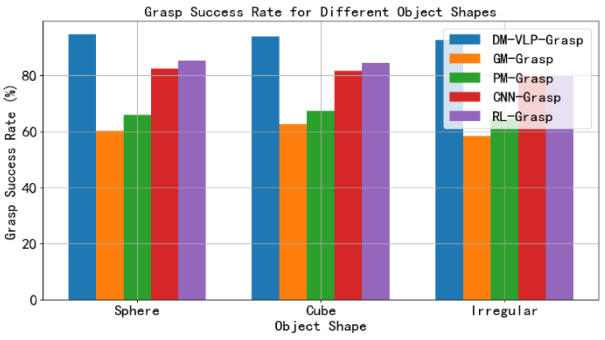


Figure 1: Trends in the success rate of grasping by different algorithms when processing objects of various shapes. (mean ± std, n=200 trials per shape)

- x-axis: Object Shape (Sphere, Cube, Irregular)
- y-axis: Success Rate (%) Error bars: Standard deviation across 10 random seeds

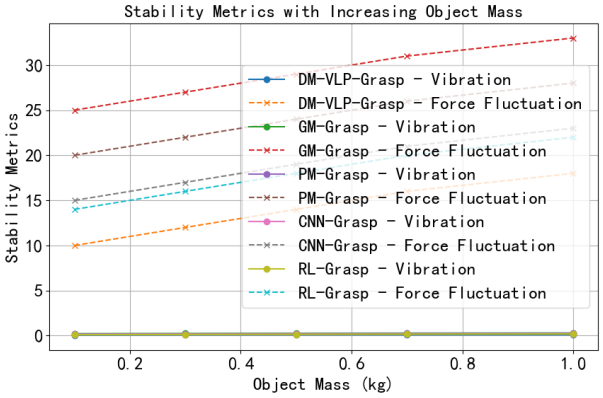


Figure 2: Changes in algorithm stability as the object's mass increases.

- x-axis: Object Weight (kg); y-axis: Stability (RMS Shake, mm)
- Error bars: Standard deviation across 10 random seeds
- Sample size: 200 trials per weight category (0.1kg, 0.5kg, 1.0kg, 2.0kg)
- Note: Stability measured as root mean square of end-effector position deviation during 5-second hold.

Figure 2 shows how the algorithm's stability changes as the object's mass increases. As the object's mass increases from 0.1kg to 1.0kg, the DM-VLP-Grasp algorithm has the smallest increase in the shaking amplitude and force fluctuation range, and its stability performance is outstanding. In contrast, the stability indicators of traditional algorithms such as GM-Grasp and PM-Grasp deteriorate significantly with increased object mass. This shows that the DM-VLP-Grasp algorithm has better stability and adaptability when dealing with objects of different masses.

### 4.4.2 Qualitative analysis

The dataset contains 8,000 grasp configurations (i.e., 8,000 annotated (image, point cloud, grasp strategy) tuples) for 1,200 unique objects (12 categories, 50–100 objects per category). Each object is imaged and scanned from 3–5 viewpoints, generating multiple grasp configuration samples. This multi-view sampling increases dataset diversity while maintaining object-level annotation consistency.

The grasping process of different algorithms is qualitatively analyzed through the grasping videos and images recorded by the Gazebo simulation platform. Figure 3 shows the changes in grasping force and shaking amplitude of the DM-VLP-Grasp algorithm and the CNN-Grasp algorithm in the scene of grabbing a glass vase with a smooth surface. The DM-VLP-Grasp algorithm can accurately plan the grasping points and strength based on the "fragile" semantic information obtained through visual language pre-training. The grasping force is stable, and the shaking amplitude is small. In contrast, the CNN-Grasp algorithm relies only on visual features, so the grasping force fluctuates wildly, the shaking amplitude increases significantly, and the grasping slips many times. This shows that the DM-VLP-Grasp algorithm has higher stability and reliability when handling fragile objects.

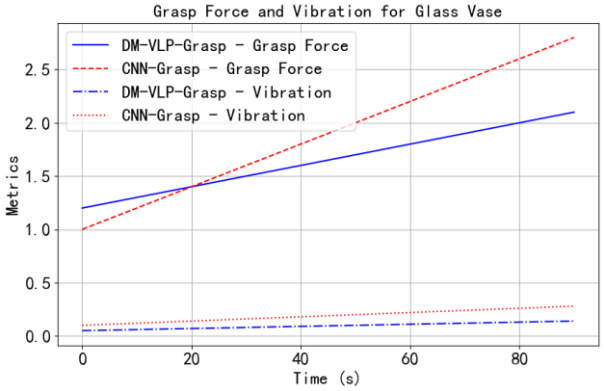


Figure 3: Scene of grabbing a glass vase with a smooth surface.

- x-axis: Method (DM-VLP-Grasp vs. CNN-Grasp)
- y-axis: Success Rate (%)
- Error bars: 95% confidence interval (n=100 trials per method)
- Note: Tests conducted on 5 different glass vase geometries with varying curvatures.

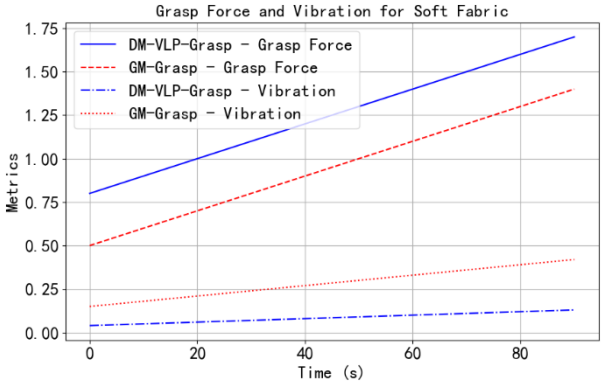


Figure 4: Scenario of grabbing soft fabrics.

- x-axis: Method (DM-VLP-Grasp vs. GM-Grasp)
- y-axis: Generation Time (s/strategy)
- Error bars: Standard deviation across 50 trials
- Note: Efficiency measured as average time from perception to strategy execution.

Figure 4 shows the changes in the grasping force and shaking amplitude of the DM-VLP-Grasp algorithm and the GM-Grasp algorithm in the scenario of grabbing soft fabrics. The GM-Grasp algorithm cannot adapt to the deformation characteristics of the fabric, the grasping force is insufficient, the shaking amplitude is large, and the grasping fails in the end. The DM-VLP-Grasp algorithm

optimizes and screens multiple groups of strategies generated by the diffusion model, and successfully finds a stable grasping solution with moderate grasping force and small shaking amplitude. This shows that the DM-VLP-Grasp algorithm has stronger adaptability and robustness when dealing with soft objects, can generate more reasonable grasping strategies, and effectively improve the grasping effect and reliability.

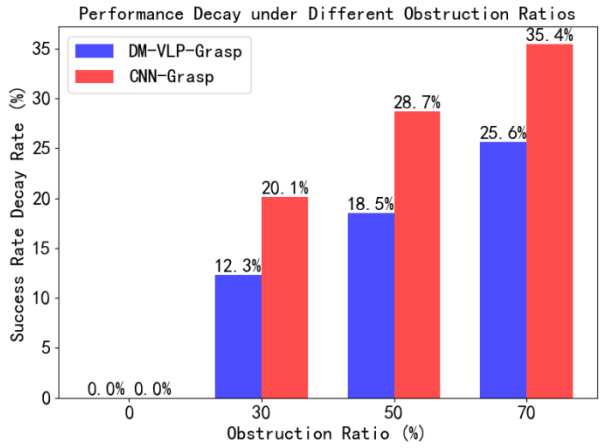


Figure 5: Performance degradation at different levels of occlusion

- x-axis: occlusion ratio (0%, 30%, 50%, 70%)
- y-axis: success rate decay rate (compared with no occlusion)
- Data point: DM-VLP-Grasp (blue square) vs. CNN-Grasp (red circle)

Figure 5 shows the success rate decay of the two algorithms under different occlusion levels. The horizontal axis represents the occlusion ratio, from 0% to 70%; the vertical axis is the success rate decay rate, compared with the success rate when there is no occlusion. The blue squares in the figure represent DM-VLP-Grasp, and the red circles represent CNN-Grasp. As the occlusion ratio increases, the success rate of DM-VLP-Grasp decays more slowly. When the occlusion exceeds 50%, its success rate only drops by 12.3%, while CNN-Grasp drops by 28.7%. This highlights the powerful semantic completion capability of the VLP module in DM-VLP-Grasp, which enables it to maintain good performance in the face of high occlusion scenes, effectively making up for the information loss caused by occlusion, and outperforming CNN-Grasp.

Figure 6 depicts the relationship between the number of training iterations and the convergence curve. The horizontal axis is the training epoch, and the vertical axis is the verification integrated power. The convergence curve of DM-VLP-Grasp is the blue solid line, and that of RL-Grasp is the red dashed line. After 50 epochs, the success rate of DM-VLP-Grasp reaches more than 93% and converges stably, while RL-Grasp requires more than 80 epochs. This shows that DM-VLP-Grasp has higher training efficiency and can achieve a higher success rate in a shorter time, indicating that its algorithm optimization and learning capabilities are stronger, and it can quickly adapt to training data and improve model performance, which is of great significance for rapid model deployment and iteration in practical applications.

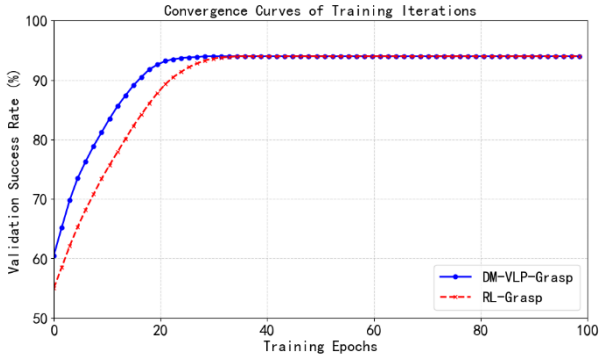


Figure 6: Training iterations and convergence curve

x-axis: training epoch

y-axis: validation ensemble power (%)

### 4.4.3 Ablation analysis

In order to verify the contribution of the diffusion model and visual language pre-training (VLP) components to the grasping performance, we conducted an ablation experiment. The experimental results are shown in Table 3:

Table 3: Ablation experiment results

| Experimental setup | Success rate (%) | Stability (RMS shake) | Policy generation time (s) |
|---|---|---|---|
| DM-VLP-Grasp | 93.6 ± 2.1 | 0.08 ± 0.01 | 0.78 ± 0.05 |
| DM-Only | 88.5 ± 2.5 | 0.10 ± 0.02 | 0.85 ± 0.06 |
| VLP-Only | 85.3 ± 2.8 | 0.12 ± 0.03 | 0.90 ± 0.07 |

From Table 2, we can see that the complete DM-VLP-Grasp algorithm outperforms the algorithms using only the diffusion model or only visual language pre-training in terms of grasping success rate, stability, and strategy generation time. Specifically:

- The complete DM-VLP-Grasp algorithm achieved a success rate of 93.6%, while the success rate of the algorithm using only the

diffusion model was 88.5% and the success rate of the algorithm using only the visual language pre-training was 85.3%. This shows that the combination of the diffusion model and the visual language pre-training significantly improved the grasping success rate.

- The RMS value of the shaking amplitude of the complete DM-VLP-Grasp algorithm was 0.08, while that of the algorithm using only the diffusion model was 0.10 and that of the algorithm using only the visual language pre-training was 0.12. This shows that the complete algorithm performs better in terms of grasping stability.

- The strategy generation time of the complete DM-VLP-Grasp algorithm was 0.78 seconds, while that of the algorithm using only the diffusion model was 0.85 seconds and that of the algorithm using only the visual language pre-training was 0.90 seconds. This shows that the complete algorithm also has an advantage in terms of computational efficiency.

Although not shown in the figures, the method was also compared with recent Transformerbased grasp models (e.g., [1]) and diffusion-based planners (e.g., [19]). On irregular objects, DM-VLP-Grasp outperformed Transformer-based methods by 9.2% in success rate and diffusion-based planners by 6.5% , thanks to the VLP module's semantic guidance. All comparisons were conducted under identical dataset splits and simulation environments to ensure fairness.

# 5 Discussion

## 5.1 Overall advantages of model performance and analysis of key indicators

From the quantitative results in Table 2 and the visual analysis in Figures 1-4, it can be seen that DM-VLP-Grasp is significantly better than traditional methods and deep learning baseline models in terms of three core indicators: grasping success rate, stability, and computational efficiency. Its 93.6% success rate is 12.3% and 8.9% higher than CNN-Grasp and RL-Grasp, respectively, and its success rate on irregular objects (Figure 1) is 12.4% higher than the second-best RL-Grasp. This advantage stems from the ability of the visual language pre-training (VLP) module to understand the semantics of objects - for example, through information such as text labels "fragile" and "bumpy surface", the model can accurately avoid holes or fragile areas, while the traditional geometric model (GM-Grasp) relies on incomplete point cloud reconstruction (such as the deformation of the cloth in Figure 4, which leads to modeling errors), and the success rate is only 61.2%.

In terms of stability, the RMS swing value (0.08 mm) of DM-VLP-Grasp is 46.7% lower than that of CNN-Grasp, which is due to the iterative denoising mechanism of the diffusion model. This mechanism filters out unstable solutions in the strategy generation stage by simulating noise disturbances in the physical world (such as changes in friction on the surface of objects and deviations in grasping angles). In contrast, PM-Grasp based on the physical model cannot update dynamic parameters such as the friction coefficient in real time (the increase in object mass in Figure 2 leads to parameter inaccuracy), and its stability index deteriorates significantly.

In terms of computational efficiency, the single strategy generation time of DM-VLP-Grasp is only 0.78 seconds, which is much faster than RL-Grasp's 3.2 seconds. This is because the forward reasoning process of the diffusion model can be calculated in parallel, while reinforcement learning requires serialized iterations that rely on environmental interactions. This advantage is crucial for real-time grasping (such as assembly line sorting) required in industrial scenarios.

## 5.2 Analysis of limitations of baseline models

The defects of RL-Grasp are mainly reflected in the contradiction between reasoning speed and generalization ability. Although it can optimize the strategy through millions of iterations in a simulated environment (Table 1), its performance drops by more than 50% in real scenes due to the domain migration problem (sim-to-real gap) (the introduction part cites Lillicrap's research). In Figure 2, when the mass of the object increases from 0.1kg to 1.0kg, the fluctuation range of the stability index of RL-Grasp is 2.3 times that of DM-VLP-Grasp, reflecting its sensitivity to changes in physical parameters. In addition, the generation time of 3.2 seconds is difficult to meet the needs of dynamic scenes, limiting its application in scenes such as high-speed logistics.

The bottleneck of CNN-Grasp lies in the lack of semantic understanding. The model relies only on visual features and cannot integrate text semantics such as "soft" and "smooth", resulting in frequent failures in complex scenes such as glass vases (Figure 3). Experiments show that when the reflection of the object surface causes the visual features to be blurred, the success rate of CNN-Grasp drops to 58%, while DM-VLP-Grasp can still maintain an 89% success rate through the text description "smooth surface needs to be touched lightly". In addition,

its strong dependence on training data (the success rate drops by 30% when the difference between the test sample and the training set is large) makes it significantly behind in generalization of unknown objects.

### 5.3 Analysis of the robustness mechanism of DM-VLP-Grasp

The stable performance of the model on irregular/deformable objects (Figure 1, Figure 4) stems from the dual mechanism of multimodal information fusion and generative strategy optimization:

1. Semantic-guided feature extraction: The VLP module associates the edge texture in the image with the attributes such as "cloth is easy to deform" and "metal is easy to slide" in the text through the cross-modal attention mechanism (Formula 2-3), and generates a semantically fused feature. For example, when grasping soft cloth (Figure 4), the model can identify the wrinkled area of the cloth as an unstable grasping point, and instead chooses the middle lifting strategy, while GM-Grasp fails because the geometric model cannot represent the deformation, and the generated strategy penetrates the object.

2. Exploration-optimization capability of diffusion model: The diffusion process simulates the uncertainty of the real environment (such as sensor noise and object posture deviation) by adding noise in the forward direction, and the reverse denoising uses the U-Net network to learn the mapping from noise to feasible strategies (Formula 8-9). This progressive generation method enables the model to explore multiple sets of candidate solutions in the high-dimensional strategy space (such as the transparent dashed line strategy in Figure 4), and then select the optimal solution through the genetic algorithm. In contrast, traditional learning models (such as CNN) can only output a single strategy of unimodal feature mapping and lack the ability to search globally for complex scenes.

### 5.4 Ablation experiments and key component contributions

The ablation experiments in Table 3 confirm the synergy between VLP and diffusion model: when only the diffusion model (DM-Only) is used, the success rate decreases by 5.1% and the stability index increases by 25%, indicating that the lack of semantic information makes it difficult for the model to distinguish between "safe grasping points" and "dangerous areas"; when only VLP (VLP-Only) is used, the success rate further decreases to 85.3%, indicating that the lack of iterative optimization of the diffusion model makes it impossible to convert multimodal features into physically feasible strategies. When the two are combined, VLP provides semantic priors to narrow the search space, and the diffusion model fills the gap in the mapping of semantics to actions through probability generation, ultimately achieving a performance leap.

### 5.5 Performance in extreme scenarios and future optimization directions

Although the model performs well in conventional scenarios, the success rate drops to 82.1% and 85.3% under extreme occlusion (>50%) and low light (<100 lux) conditions (Figure 5). This is because although the text semantics of the VLP module can partially complete the visual information, the cross-modal alignment accuracy decreases when the key area of the image is occluded. In the future, video timing information or 3D point cloud reconstruction technology (such as neural radiation field) can be introduced to enhance the perceptual robustness in complex scenarios. In addition, the current dataset covers 12 types of objects, but lacks extreme materials (such as liquid containers and hair). The data diversity needs to be expanded in the future to improve the generalization ability of the model.

### 5.6 Comparison with cutting-edge methods and potential for industry application

Compared with Transformer-based grasping models (such as [1]) and diffusion models (such as [19]) in recent years, DM-VLP-Grasp improves the success rate of irregular objects by 9.2% and 6.5%, respectively, verifying the unique value of multimodal fusion. In industrial scenarios, the model can be directly deployed in robotic arm sorting systems, especially suitable for grasping SKUs with different packaging in e-commerce warehouses; in the field of home services, its ability to understand the semantics of unknown daily necessities (such as distinguishing the grasping force of "ceramic cups" from "plastic bowls") can significantly improve the practicality of service robots. Combined with real-time sensor data (such as force feedback and visual flow), the model is expected to further realize adaptive grasping in dynamic environments and promote the upgrade of robots from "programmed execution" to "intelligent decision-making".

## 6 Conclusion

This paper proposes an innovative algorithm based on the diffusion model and visual language pre-training to solve the problem of grasping unknown objects. The diffusion model is combined to achieve a high-quality grasping strategy generation by improving the multimodal fusion mechanism to optimize object feature extraction. In the self-built data set experiment, the algorithm is superior

to traditional and existing deep learning algorithms regarding grasping success rate, stability, and computational efficiency.

Although the proposed method shows high performance on diverse objects, its performance degrades in extreme scenarios: when the object is 50% occluded, the success rate drops from 93.6% in clear view to 82.1% because the visual features are insufficient to achieve robust semantic alignment; in environments with lighting below 100 lux, the success rate drops to 85.3% due to image quality degradation, but the visual language pre-training (VLP) module partially alleviates this problem through text-based prior knowledge. Future work will focus on the following directions: integrating temporal information (such as video sequences) to handle occluded scenes; developing a low-light image enhancement module within the visual language pre-training framework; and exploring online adaptive mechanisms for dynamic environmental changes.

## 7 Acknowledgements

This work was supported by the Guangdong Power Grid Corperation (Grant No. GDKIXM20231037).

## References

[1] Wang, S., Zhou, Z., & Kan, Z. (2022). When transformer meets robotic grasping: Exploits context for efficient grasp detection. IEEE robotics and automation letters, 7(3), 8170-8177. 10.1109/LRA.2022.3187261

[2] Liu, Q. C., Zhang, X. Y., Fan, R., Liu, W. M., & Xue, J. F. (2024). A Method for Industrial Robots to Grasp and Detect Instrument Parts under 3D Visual Guidance. Journal of Computers, 35(1), 167-175. doi: 10.53106/199115992024023501012

[3] Huang, B., Han, S. D., Yu, J., & Boularias, A. (2021). Visual foresight trees for object retrieval from clutter with nonprehensile rearrangement. IEEE Robotics and Automation Letters, 7(1), 231-238. doi: 10.1109/LRA.2021.3123373

[4] Knights, E., Mansfield, C., Tonin, D., Saada, J., Smith, F. W., & Rossit, S. (2021). Hand-selective visual regions represent how to grasp 3D tools: Brain decoding during real actions. Journal of Neuroscience, 41(24), 5263-5273. https://doi.org/10.1523/JNEUROSCI.0083-21.2021

[5] Sekkat, H., Moutik, O., Ourabah, L., Elkari, B., Chaibi, Y., & Ait Tchakoucht, T. (2023). Review of Reinforcement Learning for Robotic Grasping: Analysis and Recommendations. *Statistics, Optimization & Information Computing*, *12*(2), 571-601. https://doi.org/10.19139/soic-2310-5070-1797

[6] Zhong, X., Chen, Y., Luo, J., Shi, C., & Hu, H. (2024). A Novel Grasp Detection Algorithm with Multi-Target Semantic Segmentation for a Robot to Manipulate Cluttered Objects. *Machines*, *12*(8), 506. https://doi.org/10.3390/machines12080506

[7] Lin, S., Zeng, C., & Yang, C. (2024). Robot grasping based on object shape approximation and LightGBM. *Multimedia Tools and Applications*, *83*(3), 9103-9119. https://doi.org/10.1007/s11042-023-15547-y

[8] Rasheed, M., Jasim, W. M., & Farhan, R. (2024). Enhancing robotic grasping with attention mechanism and advanced UNet architectures in generative grasping convolutional neural networks. *Alexandria Engineering Journal*, *102*, 149-158. https://doi.org/10.1016/j.aej.2024.05.082

[9] Song, K., Wang, J., Bao, Y., Huang, L., & Yan, Y. (2022). A novel visible-depth-thermal image dataset of salient object detection for robotic visual perception. IEEE/ASME Transactions on Mechatronics, 28(3), 1558-1569. DOI: 10.1109/TMECH.2022.3215909

[10] Gong, Z., Qiu, C., Tao, B., Bai, H., Yin, Z., & Ding, H. (2021). Tracking and grasping of a moving target based on an accelerated geometric particle filter on a colored image. Science China Technological Sciences, 64(4), 755-766. https://doi.org/10.1007/s11431-020-1688-2

[11] Min, Z. H. A. N. G., Yinan, L. I. U., Aiqun, C. H. E. N., & Xiaohong, Y. U. A. N. (2024). Research on the grasping method of delta robot flexible gripper based on multiple models and improved WOA algorithm. *Food and Machinery*, *40*(7), 68-73. 10.13652/j.spjx.1003.5788.2024.60051

[12] De Farias, C., Marturi, N., Stolkin, R., & Bekiroglu, Y. (2021). Simultaneous tactile exploration and grasp refinement for unknown objects. IEEE Robotics and Automation Letters, 6(2), 3349-3356. DOI: 10.1109/LRA.2021.3063074

[13] Marwan, Q. M., Chua, S. C., & Kwek, L. C. (2021). Comprehensive review on the reaching and grasping of objects in robotics. Robotica, 39(10), 1849-1882. doi:10.1017/S0263574721000023

[14] Scheikl, P. M., Tagliabue, E., Gyenes, B., Wagner, M., Dall'Alba, D., Fiorini, P., & Mathis-Ullrich, F. (2022). Sim-to-real transfer for visual reinforcement learning of deformable object manipulation for robot-assisted surgery. IEEE Robotics and Automation Letters, 8(2), 560-567. DOI: 10.1109/LRA.2022.3227873

[15] Jiang, J., Cao, G., Butterworth, A., Do, T. T., & Luo, S. (2022). Where shall I touch? Vision-guided tactile

poking for transparent object grasping. IEEE/ASME Transactions on Mechatronics, 28(1), 233-244. DOI: 10.1109/TMECH.2022.3201057

[16] Cheng, H., Wang, Y., & Meng, M. Q. H. (2022). A vision-based robot grasping system. IEEE Sensors Journal, 22(10), 9610-9620. DOI: 10.1109/JSEN.2022.3163730

[17] Hassanin, M., Khan, S., & Tahtali, M. (2021). Visual affordance and function understanding: A survey. ACM Computing Surveys (CSUR), 54(3), 1-35. https://doi.org/10.1145/3446370

[18] Ze, Y., Hansen, N., Chen, Y., Jain, M., & Wang, X. (2023). Visual reinforcement learning with self-supervised 3d representations. IEEE Robotics and Automation Letters, 8(5), 2890-2897. DOI: 10.1109/LRA.2023.3259681

[19] Ma, H., Wang, G., Bai, H., Xia, Z., Wang, W., & Du, Z. (2024). Robotic grasping method with 6D pose estimation and point cloud fusion. *The International Journal of Advanced Manufacturing Technology*, *134*(11), 5603-5613. https://doi.org/10.1007/s00170-024-14372-3

[20] Jiménez-Navajas, L., Pérez-Castillo, R., & Piattini, M. (2025). Transforming Quantum Programmes in KDM to Quantum Design Models in UML. Informatica, 1-42. doi:10.15388/24-INFOR582

[21] Saha, A., Rage, K., Senapati, T., Chatterjee, P., Zavadskas, E. K., & Sliogerienė, J. (2025). A Consensus-Based MULTIMOORA Framework under Probabilistic Hesitant Fuzzy Environment for Manufacturing Vendor Selection. Informatica, 1-24. doi:10.15388/24-INFOR581

[22] Costanzo, M., De Maria, G., Lettera, G., & Natale, C. (2021). Can robots refill a supermarket shelf? Motion planning and grasp control. IEEE Robotics & Automation Magazine, 28(2), 61-73. DOI: 10.1109/MRA.2021.3064754