Autonomous Recognition Grasping Robot Based on Speech Input

Yaoyu Cheng*, Lijie Sheng†, Zishang Zhang‡, Chenbo Bao§ and Zhongtang Zhang¶
*Southern University of Science and Technology
Email: 12010922@mail.sustech.edu.cn
†Southern University of Science and Technology
Email: 12011127@mail.sustech.edu.cn
‡Southern University of Science and Technology
Email: 12012305@mail.sustech.edu.cn
§Southern University of Science and Technology
Email: 12012309@mail.sustech.edu.cn
¶Southern University of Science and Technology
Email: 12012330@mail.sustech.edu.cn

***Abstract*—This paper studies the task of any objects grasping from the known categories by free-form language instructions. This task demands the technique in computer vision, natural language processing, and robotics. We bring these disciplines together on this open challenge, which is essential to human-robot interaction. Critically, the key challenge lies in inferring the category of objects from linguistic instructions and accurately estimating the 6-DoF information of unseen objects from the known classes. In contrast, previous works focus on inferring the pose of object candidates at the instance level. This significantly limits its applications in real-world scenarios. In this paper, we propose a language-guided 6-DoF category-level object localization model to achieve robotic grasping by comprehending human intention. To this end, we propose a novel two-stage method. Particularly, the first stage grounds the target in the RGB image through language description of names, attributes, and spatial relations of objects. The second stage extracts and segments point clouds from the cropped depth image and estimates the full 6-DoF object pose at category-level. Under such a manner, our approach can locate the specific object by following human instructions, and estimate the full 6-DoF pose of a category-known but unseen instance which is not utilized for training the model.**

I. Introduction

Understating natural language instruction is an essential skill for domestic robots, releasing humans from pre-defining a specific target for robot grasping by programming. This inspires the task of making robots understand human instructions. In this task, the robot demands to localize the target object by parsing the names, potential attributes, and spatial relations of objects from the language description. Thus it is non-trivial to make robotic grasping by linguistic description, as this task requires mature techniques from Computer Vision (CV), Natural Language Processing (NLP), and robotics. In this paper, we bring these disciplines together on this open challenge, which is essential to human-robot interaction.

To summarize, in this paper, we propose a category-level 3D object localization model to grasp unseen instances via natural language description. Take an RGB-D image and a natural language description as input, our goal is to infer the 6-DoF pose of the most likely object that matches the description. Firstly, The speech input is processed by NLP method, and the characteristic labels of the object description are extracted. Then, the classification and definition of the objects placed in the big world are defined by computer vision. Second, match the two labels to frame the object we expect to capture on the RGB image. Then, CNN is used to process RGB images to obtain the world coordinates of the object. Finally, in order to ensure the smooth and integrity of the overall process, we will use forward kinematics solution to control the manipulator to grasp it specifically.

II. Problem Statement

The key challenge lies in inferring the category of objects from linguistic instructions, and accurately estimating the 6-DoF information of unseen objects from the known classes. Specifically, the vanilla object pose estimation approaches[13][3][15] attempt to estimate the poses of objects from the image, while we aim at locating specific objects using a natural language description. Here we employ convolutional neural networks (CNN) and connectionist temporal classification (CTC) to parse the linguistic instructions and generate features related to the features of the captured objects. Furthermore, the other thing that we focus on is, how do we match the object labels that we get with NLP to the actual objects that are laid out in the world. We decided to use the combination of yolo v5 and computer vision to classify all objects in the world environment, that is, give them different labels that conform to objective basis, and then match this label with the voice label obtained through NLP, so as to obtain the actual information of the object we want to capture.

III. literature review

A. Real-Time Grasp Detection Using Convolutional Neural Networks

Redmon J et al.[11] proposed in 2015 a convolutional neural network model ”tailored” for the grasping problem of target

objects in depth images. Compared with traditional methods, this grasping model can be well extended to new objects, and only a single view is needed, rather than a complete 3D model. The main idea of this convolutional neural network model is to apply a single neural network to the whole image to predict the grasping coordinates. The network performance is quite excellent because it avoids the computational cost of running a small scale classifier on a small area of the image many times, and turns to the global grasping prediction of the complete image of an object.

Wu YX et al.[16] presented in 2022 a novel anchor-free grasp detector based on fully convolutional network for detecting multiple valid grasps from RGB-D images in real time. Grasp detection is formulated as a closest horizontal or vertical rectangle regression task and a grasp angle classification task. By directing predicting grasps at feature points, our method eliminates the predefined anchors that commonly used in prior methods, and thus anchorrelated hyperparameters and complex computations are avoided. For suppressing ambiguous and low-quality training samples, a new sample assignment strategy that combines center-sampling and regression weights is proposed.

B. Chinese Speech Recognition System

The commonly used Chinese speech recognition process is divided into four steps: feature extraction, matching the speech spectrogram and corresponding pinyin using acoustic models, and decoding the Chinese text corresponding to pinyin.

The most important part of this is feature extraction. Typically, feature extraction is given to human ear auditory structures, As we all know, human speech is produced through the initial sound produced by the vocal apparatus in the body, which is filtered by the shape of the vocal tract formed by other objects, including the tongue and teeth, to produce a wide variety of speech sounds. Traditional speech feature extraction algorithms are based on this, and with some digital signal processing algorithms, they are able to include the relevant features more accurately, thus helping the subsequent speech recognition process. Common speech feature extraction algorithms include MFCC, FBank, LogFBank, etc. The thematic flow of these programs is shown in Figure 1.

The core step in the above workflow is the Mel filter, which mimics the structure of the human cochlea, and this filter set consists of 20-40 triangular filters. The center frequency and bandwidth of each triangular filter are determined according to the Mel scale, a nonlinear frequency scale based on the perceived pitch of the human ear. The response function of each filter is convolved with the spectrogram to obtain the output of each filter in the frequency domain. This output represents the intensity of the sound in that frequency band, which is equivalent to dividing the original signal into a number of band signals of different frequencies. The logarithm of each filter output is taken as the feature vector. The reason for this is that the human ear perceives on a logarithmic scale, so the logarithmic transform can better simulate the human ear's perception of sound.[12]

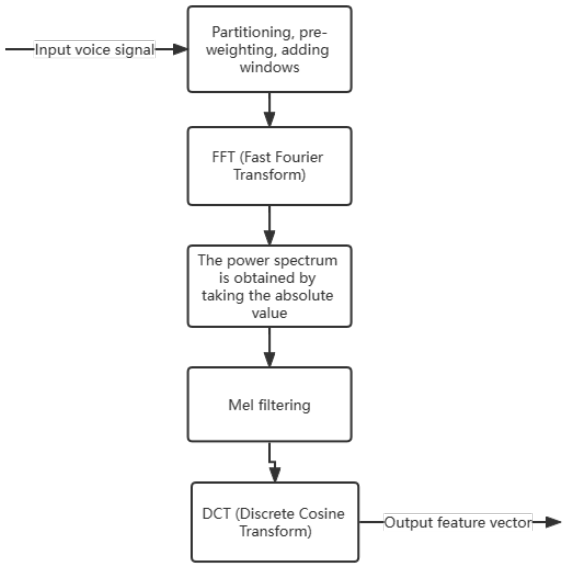

Fig. 1. Flow chart of common feature extraction algorithms.

However, with the development of neural networks. To improve the speech recognition rate, it is necessary to overcome the variety of diversity faced by the speech signal, including the diversity of speakers (the speaker itself, as well as the speakers), the diversity of the environment, etc. A convolutional neural network provides translational invariant convolution in time and space, and by applying the idea of convolutional neural network to acoustic modeling of speech recognition, the invariance of convolution can be used to overcome the diversity of speech signal itself. From this point of view, it can be considered that the entire time-frequency spectrum obtained from the analysis of the speech signal is treated like an image, and the deep convolutional network widely used in images is used for its recognition.

Based on these information, we try to implement Chinese speech recognition using convolutional neural network as the core network. In this project, the acoustic model is trained by using Convolutional Neural Network (CNN) and Connectivity Temporal Classification (CTC) methods to transcribe sounds into Chinese pinyin by using a large Chinese speech dataset, and to convert the pinyin sequences into Chinese text by language model.

C. 6-DOF Manipulator Grasps

The key words of this project are: manipulator simulation, manipulator grasping, kinematics solving, trajectory planning, deep learning.

1) The mainly basic theory: 1. Trajectory planning and motion simulation of six-axis manipulator based on matlab. In this respect, we have read several articles widely cited in China, and carried out simulation learning of four and six degrees of freedom manipulator on MATLAB platform.

2. ROS moveit! Robotic arm control and grasping. Since ROS system is the integration of the robot operating platform,

most simulation and learning of robotic arms are carried out on this platform. In addition to publishing position and speed instructions with joint publisher, moveit can also be used to directly carry out trajectory planning and mode selection, as referred to in the two domestic articles here[8].

2) Machine learning section: Training with different algorithms: - 1. Object position and terminal attitude selection.

This part has been mentioned in the part of visual information processing, but for the trajectory planning of the robot arm, we must pay attention to the position and pose of the grasping object. Here, we refer to a relatively basic article in China, which firstly uses the object detection algorithm based on deep learning to detect the object in the image and record the category and position of the object. [6], according to the classification detection results, the manipulator grasping method based on deep learning is used for grasping position learning. At the same time, we selected a relatively new foreign article, which mainly studied the method of self-supervised learning, based on the random sampling principle of OMPL motion trajectory plug-in principle and the use of the method, and complex environment considering the dynamic system of KPIECE algorithm control effect is excellent.[9]

2. Trajectory planning.

Optimization Methods	***Elapsed Time (sec)***
NSGA-II	100.92
GA	108.82
ABC	112.63
PSO	184.78

Fig. 2. Time spent in different methods

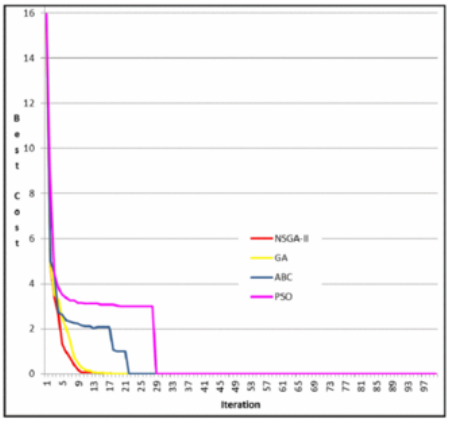

Fig. 3. The results of the best cost of distance objective function of nsga-ii, ga, abc, and pso algorithms

In keeping with the rapid development of computer technology, we looked up the latest literature as far as we could. Among them, non-dominated sorting genetic algorithm (NSGA-II), genetic algorithm (GA), artificial bee colony algorithm (ABC) and particle swarm optimization algorithm (PSO) are compared and studied to optimize point-to-point motion planning, and study the motion trajectory planning of robot arms[4].

IV. Technical Approach

A. Real-Time Grasp Detection Using Convolutional Neural Networks

1) The five dimensional representation for robotic grasps: For the grasping problem of target objects, the five-dimensional representation for robot grasping proposed by Lenz et al. [7] is generally adopted in the field of robotics. This representation gives the location and orientation of a parallel plate gripper before it closes on an object. Ground truth grasps are rectangles with a position, size, and orientation:

$$g = (x, y, \theta, h, \omega)$$

(x, y) represents the center of the rectangle, θ represents the deflection Angle of the rectangle with respect to the horizontal direction, h and ω Represents the height and width of the rectangle, as shown in Figure 4. The object grabbing problem using five-dimensional representation is similar to the object detection problem in computer vision except that the object grabbing direction is added.

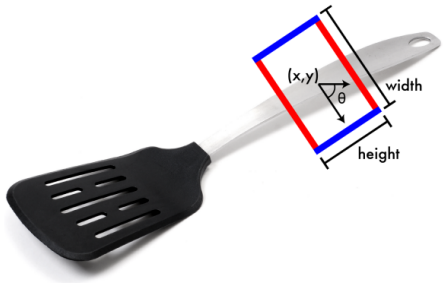

Fig. 4. A five-dimensional grasp representation, with terms for location,size, and orientation. The blue lines demark the size and orientation of the gripper plates. The red lines show the approximate distance between the plates before the grasp is executed.

2) Grasp detection with neural networks: Convolutional neural networks (CNNs) currently outperform other techniques by a large margin in computer vision problems such as classification [1] and detection [10]. We will harness the extensive capacity of a large convolutional network to make global grasp predictions on the full image of an object.

This neural network is based on AlexNet convolutional neural network model proposed by Krizhevsky et al. The network has five convolutional layers followed by three fully connected layers. The convolutional layers are interspersed with normalization and maxpooling layers at various stages. There are 6 outgoing neurons in the output layer, corresponding to the captured coordinates. Among them, four of the neurons correspond to location and height. Grasp angles are two-fold rotationally symmetric so we parameterize by using the two additional coordinates: the sine and cosine of twice the angle. A full description of the architecture can be found in Figure 5.

After obtaining the five-dimensional parameters for robot grasping through the convolutional neural network and combining with the depth information read by the depth camera,

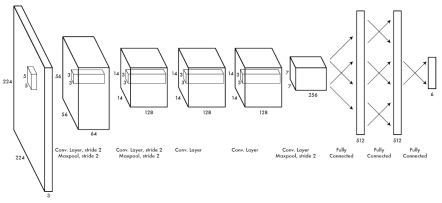

Fig. 5. The full architecture of our direct regression grasp model.

the complete grasping parameters required by the manipulator can be obtained.

B. Speech recognition system based on deep learning implementation

1) Feature Extraction: The extraction algorithm of MFCC features we mentioned above is not so much about extracting features as it is about just pre-processing the sound signal. Conventional MFCC features, after Fourier transform, are present with various types of artificially designed filters, such as Mel filters. These artificial auditory feature-based speech feature extraction are based on some a priori knowledge, for example, people are not sensitive to hearing high frequency signals, then this type of processing will cause a large loss of the speech signal in the frequency domain, especially in the high frequency region. And those traditional speech feature extraction algorithms, in order to reduce the computation of slicing operation, will also produce very large time window offset in the time domain, so it will also lead to the problem of sound information loss, especially when the speaker speaks faster.

We choose to leave further feature extraction to a subsequent neural network model. The neural network can automatically learn Mel filter-like extraction features during the training process, which often outperforms traditional feature extraction algorithms in current practical applications because it contains more information that traditional algorithms discard.

2) Acoustic models implemented by convolutional neural networks: When it comes to the application of CNN in speech recognition, it is necessary to mention CLDNN [14]. CLDNN consists of three parts.CNN and LSTM can obtain better performance improvement than DNN in speech recognition tasks. For modeling ability, CNN is good at reducing frequency domain variation, LSTM can provide long time memory, so it has wide application in time domain, while DNN is suitable for mapping features to independent space. In CLDNN, CNN, LSTM and DNN are strung together and fused into one network to obtain better performance than separate networks.The network structure diagram is shown in the figure 4.

In the past year, speech recognition has made great breakthroughs. IBM, Microsoft, Baidu and many other organizations have launched their own Deep CNN models to improve the accuracy of speech recognition.The process of trying Deep CNN is also roughly divided into two strategies: one is the acoustic model based on Deep CNN structure in HMM framework, CNN can be VGG, Residual connected CNN

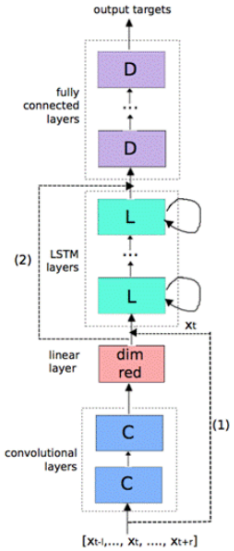

Fig. 6. CLDNN network structure diagram.

network structure, or CLDNN structure. The other one is the end-to-end structure which is very hot in the last two years, such as end-to-end modeling using CNN or CLDNN in CTC framework, or coarse-grained modeling unit techniques such as Low Frame Rate and Chain model which are proposed recently.[5]

We try to train acoustic models based on Keras and TensorFlow frameworks, using VGG's deep convolutional neural network as a network model.[2]VGG was proposed by the Visual Geometry Group at Oxford. Its main work is to demonstrate that increasing the depth of the network can affect the final performance of the network to some extent.

In brief, in VGG, three 3x3 convolutional kernels are used instead of 7x7 convolutional kernels, and two 3x3 convolutional kernels are used instead of 5x5 convolutional kernels, the main purpose of this is to improve the depth of the network while guaranteeing to have the same perceptual field, and to improve the neural network to some extent. 5x5 convolution is viewed as a small fully connected network sliding in the 5x5 region, we can first convolve with a 3x3 convolution filter, and then connect this 3x3 convolution output with a fully connected layer, which we can also see as a 3x3 convolution layer. This way we can cascade (superimpose) two 3x3 convolutions instead of one 5x5 convolution. With such a network we try to construct the prediction of speech spectrum features to pinyin.

3) CTC decoding: In the output of the acoustic model of speech recognition system, it often contains a large number of consecutive repetitive symbols, so we need to merge consecutive identical conformations into the same symbol and then remove the silence separator marker to get the final actual sequence of phonetic symbols of speech.

The core of CTC sorting is the introduction of the space character to solve the problem of repetition in the correspond-

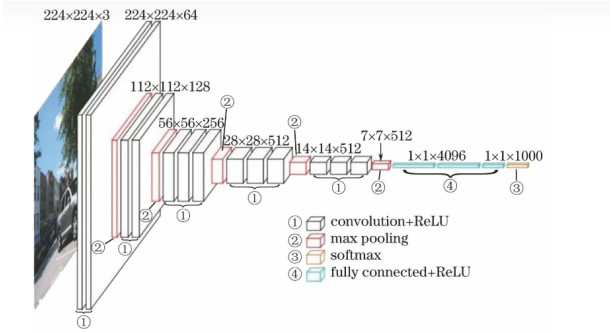

Fig. 7. VGG network structure diagram.

ing characters, while counting all the syllable combinations that can form a word and accumulating the probability to arrive at the most likely result.

The CTC calculation is based on the following equation:

$$p(Y \mid X) = \sum_{A \in \mathcal{A}_{X,Y}} \prod_{t=1}^{T} p_t\left(a_t \mid X\right)$$

Based on the designed directed graph showed in Figure 6, we can obtain all possible combinations of a sequence and calculate the probability of the corresponding sequence, and the maximum probability is the phonetic output of our corresponding speech.

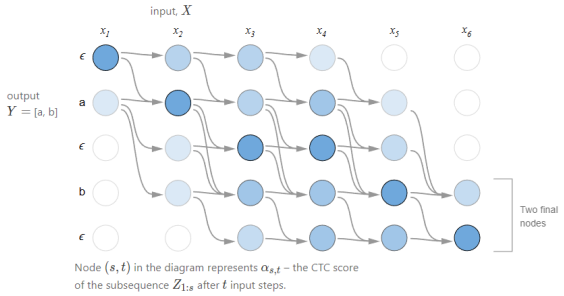

Fig. 8. CTC directed graphs.

4) Statistical language models: from Chinese pinyin to text: Theoretically, if S is a meaningful sentence consisting of a sequence of words w1,w2,...wn (n is the length of the sentence), then the probability that the text S holds, i.e., the probability P(S), is the probability that the first word occurs multiplied by the probability that the second word occurs under the condition that the first word occurs, multiplied by the probability that the third word occurs under the condition that the first two words occur and then the probability of the third word occurring under the conditions of the first two words, all the way to the last word. The probability of occurrence of each word is related to all the previous words, so we have the following formula:

$P(S) = P(w_1, w_2, \ldots, w_n) = P(w_1)^* P(w_2 \mid w_1)^* P(w_3 \mid w_1, w_2) \ldots P(w_n \mid w_1, w_2, \ldots, w_{n-1})$

But such iterative probabilities are difficult to calculate, so based on Markov assumptions, the probability of the current word can have a fairly good accuracy rate if only the previous word is considered, and in practice, it is usually enough to consider the first two words, so the formula can be simplified as follows:

$P(S) = P(w_1, w_2, \ldots, w_n) = P(w_1)^* P(w_2 \mid w_1)^* P(w_3 \mid w_2) \ldots P(w_n \mid w_{n-1})$

As for the acquisition of frequency, we follow the theorem of large numbers that relative frequency is equal to probability as long as the statistics are sufficient. Based on the dataset of the corpus, we can then obtain the probability of a single word and the probability of a phrase based on a word.

As for the probability-based word selection, we choose the Viterbi algorithm to help us do dynamic planning. The Viterbi algorithm simplifies the entire process, iterating through the previous possibilities in terms of nodes, assigning the minimum value to the calculation after this node. The entire workflow is shown in the figure 7. In terms of efficiency compared to brute force traversal of all paths, the Viterbi algorithm removes the paths that do not meet the shortest path requirement when it reaches each column, greatly reducing the time complexity. Also, since all nodes are traversed, there are no incorrect solutions.We may also try to solve this problem later by adopting the idea of reverse regression for the DP problem.

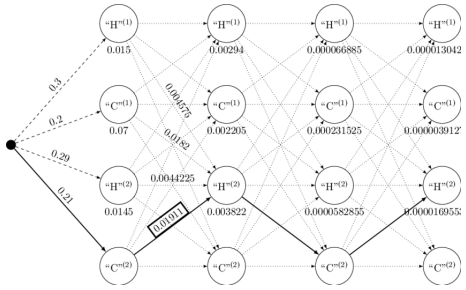

Fig. 9. Work flow of Viterbi.

C. Target detection system based on YOLOv5

1) Basic theorem: Basically, for an input image, the network would split it into an S*S grid. If the center of an object falls within one of the windows, then this window is responsible for the prediction of the object. Each window should predict M anchor boxes. Each anchor box should return 5 values, including its own position (x, y, w, h), as well as the confidence value of the prediction. Meanwhile, for each window, a number of N categories should also be predicted and labeled. So the output for the input image is a tensor of S*S(5*M+N). With the output, the redundant windows and the target windows with low probabilities would be removed. Multiple boxes and corresponding labels would be displayed in order to indicate the positions and categories of the objects.

2) Actual usage: For the section of model training, RefCOCO dataset will be used. With the configurations and parameters set, a model can be trained by running some preset

codes integrated in YOLO. By adjusting the configurations, different models will be produced and one that fits the requirements most will be used in the following steps. Set the input as the camera, and run the program pre-built by YOLO. Normally, the detected target and the probability would be shown in a window for realtime output. But our goal is to find the coordinate and the label for the objects inside the view, realtime output is not what we need primarily. So the log file is essential. A program is written to read the position and the corresponding label in the log file, and send them out for further usages in coordinate transformation.

D. Mechanical arm grasping process

1) Brief description of this part: Firstly, the premise of this part is to set the structural parameters and the forward and inverse kinematics of the manipulator. Then we need to be clear that we can process information and target for visual input. The most important thing is trajectory planning: Based on the interpolation method (matlab set parameters and calculate) teaching method (moveit) learned before, learn and try the motion trajectory plug-in of OMPL based on random sampling principle and the KPIECE algorithm for considering the dynamic system in complex environment for control, output information images, compare and evaluate the application scope of each scheme. Finally, we may try the deep learning methods in this part. Overall, it is to conduct information processing based on the deep neural network model of "knowing the target information, selecting the grasping position and terminal attitude, and carrying out path planning", so as to improve the grasping success rate and task completion degree.

2) The technology corresponding to the step: -

1. Structural parameters and forward and inverse kinematics of the manipulator: tested in different simulation environments. (matlab→ros→robosuite)

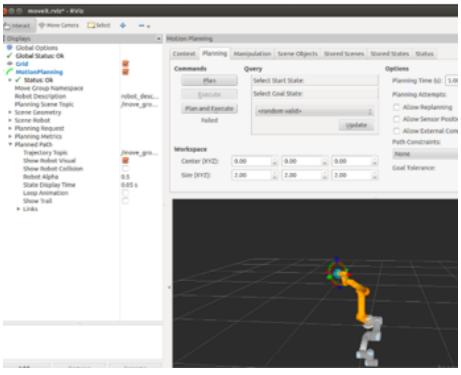

Fig. 10. A simulation in ros moveit!

2. Information input: Feature extraction, position and terminal attitude calculation (classification detection, linear regression/nearest neighbor algorithm, self-supervised learning)

3. Trajectory planning capture:

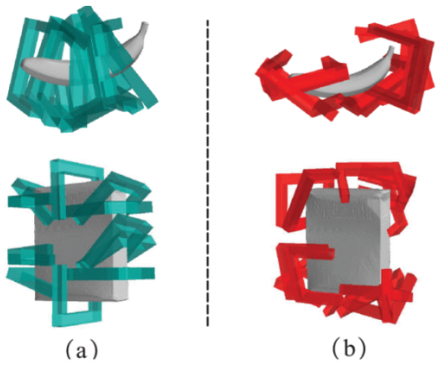

Fig. 11. Illustration of grasp pose classification.

a. Select the path points and use the polynomial function of higher order to solve

b. Learn the principle and usage of OMPL trajectory plug-in c based on random sampling principle. [6]

c. Considering the complex environment, the KPIECE algorithm of the dynamic system is controlled, and the information image is output to evaluate the scope of application of each scheme.[9]

3) Reference learning materials: Self-supervised learning, non-dominated sorting genetic algorithm (NSGA-II), genetic algorithm (GA), artificial bee colony algorithm (ABC), particle swarm optimization algorithm (PSO) and other schemes optimized point-to-point motion planning, and studied the motion trajectory planning of robot arms.[4])

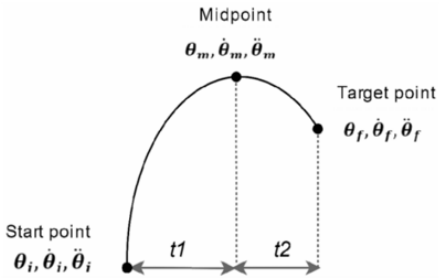

Fig. 12. Motion planning strategy from start point

V. Preliminary Results

In terms of using vision algorithms for grasping parameter calculations, open source neural network algorithms corresponding to the papers have been found and tested. And in terms of simulators (Robosuite, Webots), API interfaces corresponding to deep cameras have been found.

A. Mechanical arm simulation clamping part

1. Forward and inverse kinematics simulation was completed for ur5 robot arm of the squadron of Matlab toolbox

2. Reviewed the model import and configuration of ros melodic and rviz gazebo simulation, and moveit! Motion planning

3. Updated the Ubuntu version and installed the robosuite simulation environment (learned the structure and content of

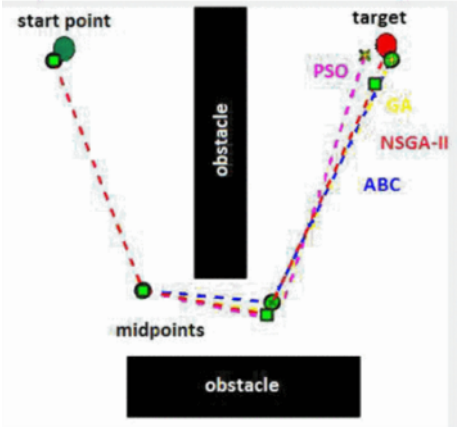

Fig. 13. Comparison of path planning between nsga-ii, abc, and pso algorithm simulation results

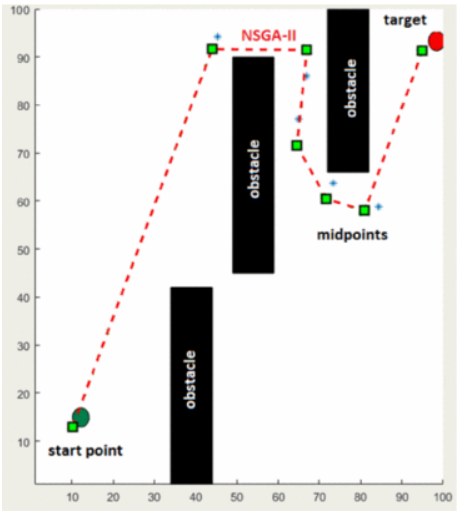

Fig. 14. Nsga-ii optimization trajectory planning simulation result

demo file, found the operation port of demo, only the official tutorial, which is quite difficult)

Fig. 15. Set up the environment in the robosuite simulation environment and open the robotic arm to operate

B. Voice Recognition

1. Successfully completed the framework built above, can well recognize the voice we input, as Fig 16. For example, we can use Chinese voice input, give me a red apple, then the system we built can successfully convert it into the corresponding Chinese, so as to ensure the smooth generation of voice feature tags later

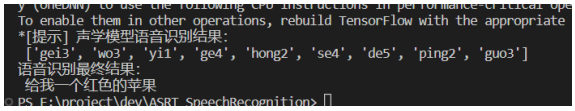

Fig. 16. Speech Recognition.

2.We are able to extract the backbone information from the translated phrases to facilitate the recognition work later. Specifically, the keyword extraction function after speech to text can be used to generate feature labels, so that the subsequent corresponding to the actual labels obtained by yolo v5 can be used to determine the objects expected to be captured

VI. Conclusion

This paper presents a novel systematic framework capable of learning 6-DoF object poses for robotic grasping from RGB-D images via language instructions. Our model estimates 6-DoF object poses at category-level. The point cloud segmentation module helps better performance in 6-DoF pose estimation. We believe our system is significant for both robotic grasping and human-robot interaction tasks.

Acknowledgments

References

[1] I. Sutskever A. Krizhevsky and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25 (2), 2012.

[2] Andrew Zisserman Andrew Zisserman. End-to-end acoustic modeling using convolutional neural networks for hmm-based automatic speech recognition. 2019. doi: 10.48550/arXiv.1409.1556.

[3] Y. Zhu C. Wang, D. Xu and R. Mart´ın-Mart´ın. "densefusion: 6d object pose estimation by iterative dense fusion". *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, page 3343–3352, 2019.

[4] Gamze Demir and Revna Acar Vural. Heuristic trajectory planning of robot manipulator. pages 222–226, 2021. doi: 10.1109/JEEIT53412.2021.9634101.

[5] Ronan Collobert Dimitri Palaz, Mathew Magimai-Doss. End-to-end acoustic modeling using convolutional neural networks for hmm-based automatic speech recognition. pages 15–32, 2019. doi: 10.1016/j.specom.2019.01.004.

[6] Cai HY DU XD. A robotic grasping method based on deep learning. *Robot*, (820-828+837), 2017. ISSN 1002-0446. doi: 10.13973/j.cnki.robot.2017.0820.

[7] H. Lee I. Lenz and A. Saxena. Deep learning for detecting robotic grasps. *The International Journal of Robotics Research*, 34(4-5):705–724, 2015. doi: 10.1177/0278364914549607. URL https://doi.org/10.1177/0278364914549607.

[8] Liang YB Meng HN. The motion planning of a six dof manipulator based on ros platform. *Journal of Shanghai Jiaotong University*, (94-97), 2016. ISSN 1006-2467. doi: 10.16183/j.cnki.jsjtu.2016.S.024.

[9] Gang Peng, Zhenyu Ren, Hao Wang, Xinde Li, and Mohammad Omar Khyam. A self-supervised learning-based 6-dof grasp planning method for manipulator. *IEEE Transactions on Automation Science and Engineering*, 19 (4):3639–3648, 2022. doi: 10.1109/TASE.2021.3128639.

[10] T. Darrell R. Girshick, J. Donahue and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. pages 580–587, 2014. doi: 10.1109/CVPR.2014.81.

[11] Angelova A Redmon J. Real-time grasp detection using convolutional neural networks. pages 1316–1322, 2015. doi: 10.1109/ICRA.2015.7139361.

[12] P. Mermelstein S. Davis. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. pages 357–366, 1980. doi: 10.1109/TASSP.1980.1163420.

[13] V. Lepetit S. Hinterstoisser and S. Ilic. "model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes". *Asian conference on computer vision*, page 548–562, 2012.

[14] Andrew Senior Tara N. Sainath, riol Vinyals. Convolutional, long short-term memory, fully connected deep neural networks. pages 19–24, 2015. doi: 10.1109/ICASSP.2015.7178838.

[15] V. Narayanan Y. Xiang, T. Schmidt and D. Fox. "posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes". 2017. doi: 1711.00199.

[16] Wu YX, Zhang FH, and Fu YL. Real-time robotic multigrasp detection using anchor-free fully convolutional grasp detector. *IEEE Transactions on Industrial Electronics*, 69(12):13171–13181, 2022. doi: 10.1109/TIE.2021.3135629.