International Journal of Humanoid Robotics

LEARNING AN IMAGE-BASED VISUAL SERVOING CONTROLLER FOR OBJECT GRASPING

Shuaijun Wang[*], Lining Sun[*], Mantian Li[*], Pengfei Wang[*], Fusheng Zha[*†], Wei Guo[*†], Qiang Li[‡]

[*]*State Key Laboratory of Robotics and System, Harbin Institute of Technology, Harbin 150080, China*

wukongwoong@gmail.com
lnsun@hit.edu.cn
lmt@hit.edu.cn
pfwang@hit.edu.cn
fushengzha@hit.edu.cn
wguo01@hit.edu.cn

[‡]*Neuroinformatics Group, Center for Cognitive Interaction Technology (CITEC), Bielefeld University, 33619 Bielefeld, Germany*

qli@techfak.uni-bielefeld.de

Received December 13, 2023
Revised December 13, 2023
Accepted December 13, 2023

Adaptive and cooperative control of arms and fingers for natural object reaching and grasping, without explicit 3D geometric pose information, is observed in humans. In this study, an image-based visual servoing controller, inspired by human grasping behavior, is proposed for an arm-gripper system. A large-scale dataset is constructed using Pybullet simulation, comprising paired images and arm-gripper control signals mimicking expert grasping behavior. Leveraging this dataset, a network is directly trained to derive a control policy that maps images to cooperative grasp control. Subsequently, the learned synergy grasping policy from the network is directly applied to a real robot with the same configuration. Experimental results demonstrate the effectiveness of the algorithm. Videos can be found at https://www.bilibili.com/video/BV1tg4y1b7Qe/.

Keywords: visual servoing; robotic grasping; robot learning.

1. Introduction

Grasping is an important area of research in intelligent robotics. A large number of studies have explored various aspects of perception, planning and control to achieve robust grasping behaviour in robots. For example, many studies have been devoted to determining a stable grasping region for complex objects, which is a difficult task influenced by factors such as object texture, shape, mass, and type [1−2]. However, in many real-world applications, robots often need to grasp simple objects (as shown in the Figure 1).

[†]Corresponding author

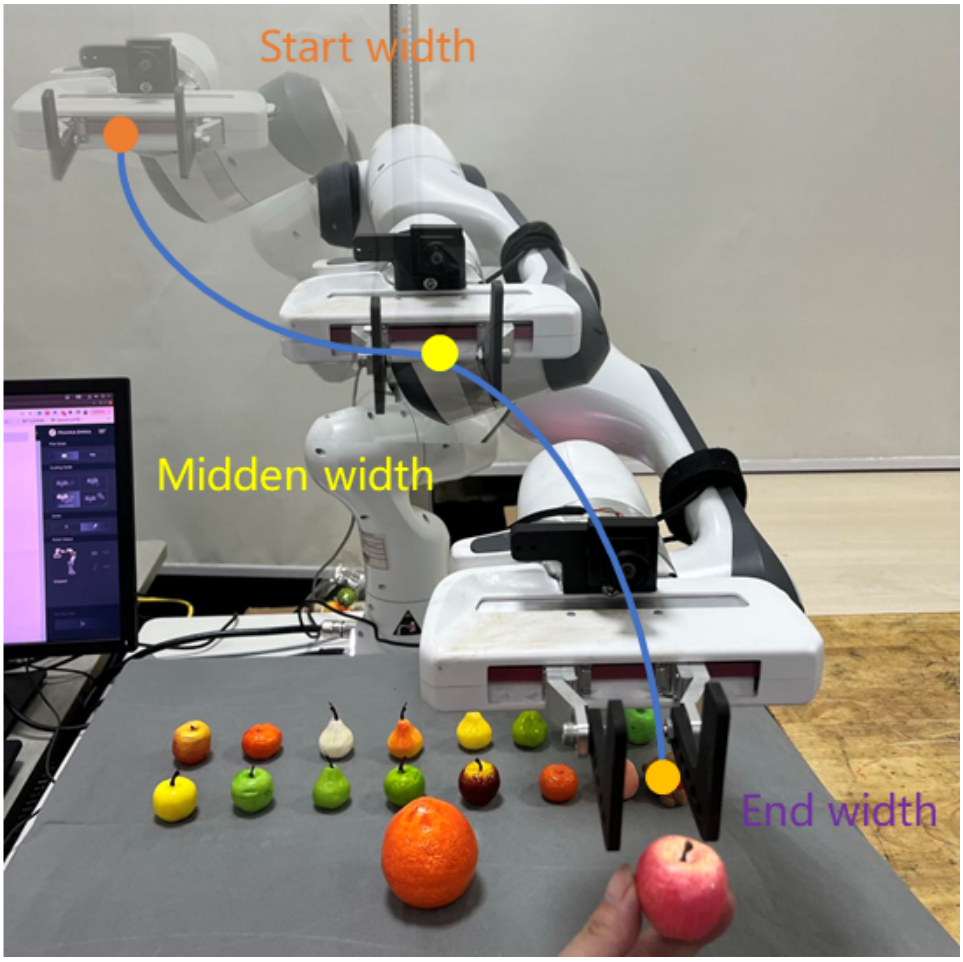


Fig. 1. Closed-loop, adaptive arm-gripper synergistic grasping in 3D space with only RGB images.

In this context, the current focus is on achieving end-to-end arm-gripper synergy between the robotic arm and gripper to approach and grasp objects in a human-like manner. Arm-gripper synergy is a novel concept introduced in this paper, which entails focusing on the coordinated motion of the hand and arm during the object grasping process. For instance, as the arm approaches an object, the fingers work in coordination to close, and as the arm moves away from the object, the fingers open. This form of coordination unifies the stages of approach and grasp into a more natural and human-like process, seamlessly integrating the approach and grasp stages into a single, more efficient process. Unlike humans, whose arm and hand movements adaptively adjust to the distance from the target object, current robots with arm and gripper lack well-researched methods to achieve this natural synergy [3].

Further, most existing grasping solutions [4] follow a three-phase framework: motion planning, grasp planning and grasp control [5]. Grasp planning defines the desired tool frame pose, motion planning generates smooth arm trajectories, and grasp control coordinates the gripper's movements to fixate the target object. However, this framework faces a number of challenges:

(a) It requires robot hand-eye calibration to localize the target in the robot coordinate system, which is a time-consuming process prone to systematic errors.

(b) Conventional methods rely on depth information to compute the pose of an object. Due to hardware limitations, depth cameras may have errors and blind spots, leading to unpredictable results.

(c) Few studies have bridged the gap between motion planning and grasping control, leading to a disconnect between arm and hand movements and unnatural grasping behaviour.

(d) In past studies, the fingers of the robotic hand would remain stationary as the arm approached an object and would only begin to close the fingers when the hand reached a specific position. This approach takes longer and produces mechanical and non-smooth grasping behaviour compared to human grasping behaviour.

Inspired by these challenges, this paper presents a learning-based framework that formulates robot approach and coordinated grasping as a visual servoing problem. Unlike traditional visual servoing methods that require depth information to compute the Jacobian matrix, this approach does not rely on depth information from the visual system, thus eliminating the problem of inaccurate or missing depth data. Instead, it generates robotic arm-gripper synergy control commands directly from pixel-level inputs, thereby facilitating natural synergistic control between the manipulator and gripper. This approach requires minimal calibration and depth information and is therefore suitable for real-time, end-to-end gripper control.

While manually separating the robotic arm and hand for gripping control is a practical engineering solution, it can lead to a non-smooth gripping process. To achieve more natural grasping, arm approach and hand control should be unified. Recent advances in robot learning, especially reinforcement learning (RL), offer the possibility of end-to-end grasping [6]. However, reinforcement learning typically requires large amounts of data, both simulated and real-world experimental, which can limit practical applications and is sensitive to the specific settings of the environment.

In this work, the proposed visual servoing algorithm builds on an expert behavioural dataset to encode spatial and temporal mappings from images into arm grasping actions. This approach achieves collaborative control of the arm and gripper and allows natural collaborative actions using only a low-cost webcam.

The main contributions of this paper are two aspects:

(1) With the use of a low-cost RGB camera, a learning-based visual servoing controller is introduced. This controller enables the robot to achieve real-time, closed-loop grasping with end-to-end arm-gripper synergy for a wide range of ob-

jects;

(2) A novel metric for evaluating the synergy motion of the robotic arm and gripper is proposed in this paper.

2. Related Work

2.1. *Arm-gripper synergetic grasping*

A wide range of research work focus to achieve a highly successful grasping rate based on data-driven methods, like Dex-Net[9], Anygrasp[10], GG-CNN[11]. By constructing a large-scale grasping data set, they underly the grasping planning policy into the data, using corresponding networks to abstract the grasping pose in the cartesian world in a supervised-learning manner. Another pipeline[12] using reinforcement learning(RL) also has an appealing performance in recent works. Yet the RL methods for robotics are commonly suffering sample efficiency, sim2real problems[13,8], especially in high-dimensional sensory data, such as images and multi-sensory combined data. In general, these two pipeline methods trends to output explicit grasp pose in position level, or image level that is calculated to explicit grasp pose by depth cameras. While few of them pay attention to the collaboration between the manipulator and gripper. Previous work[14−15] usually artificially fragmented approaching and grasping, ignoring the synergy between arm and gripper/hand. This research[3] separated the reaching and grasping manually, which can not yield natural human-like grasping and manipulation. [16] made use of RL method to learn reactive reaching and grasping skills through a well-designed reward function with arm-hand synergy, but lacking real experiments and existing potential sim2real issues.

2.2. *Grasping using RGB image only*

Most of the grasping[4,11] methods need RGB-D information to calculate the coordinated grasping pose. Some of them [17] describe grasp as a five-dimensional vector, including grasping point and orientation in the image level, which need to be recalculated to the robot grasping coordination system using depth information. The depth camera usually is needed to calculate the 6D grasping pose. But the depth camera's precision and cost need to be considered, especially in the occasion of need high-precision. The depth cameras used in industry field can be quite expensive and is not effective at close distance[11].

Using RGB images only to achieve successful grasping is challenging and promising. These work's pipeline [18−21] is first to do the pose estimation of target object using RGB only without depth information, then transfer the pose from camera coordination to the robot coordinate system using the hand-eye calibration. However, they all need a camera intrinsic to solve the PnP problem[22] to compute the 6D pose. Unlike our method, it does not need any hand-eye calibration and camera intrinsics in the grasping progress, enabling successful grasping with RGB simply.

It achieves a performance that is "Seeing is Grasping" really without any prior preparation work. All the simplicity thanks to the non-coordinated design in the grasping progress of our method. Successful grasping is defined as when the current image matches the target image.

2.3. *Grasping based on visual servoing*

Closed-loop grasping can also be commonly regarded as visual servoing[23]. Numerous works[24–26] apply visual servoing in the grasping. Yet in the grasping based on visual servoing, the jacobian matrix needs to be calculated using depth information, to map the error of image level to the camera velocity. Additionally, it requires prior or hand-made features of the image, which is not suitable for a wide range of random objects. GG-CNN[11] achieves real-time, closed-loop grasping based on a supervised learning method using an improved Cornell grasping dataset[27], but it suffers the problem of cannot obtain reliable depth information especially when the distance between the camera and target objects is close. The work[28] achieve continuous visual servoing that can grasp a wide range of objects, without precise hand-eye calibration. However, it requires a large-scale real-world dataset which is expensive.[29] is the closest method to our work, it learns a visual servoing controller as well to guide the robot to a specialized pose that they call a bottleneck point, and then it directly replays human demonstration for the next manipulation tasks. Their framework is valid only for a certain definite object and the control velocity is calculated by human-made equations while not from policy directly. In contrast, our method has an adaptive arm-gripper collaborative grasping performance with direct control velocity as output.

3. Problem Formulation

In this study, the problem of collaborative grasping of a target object by a robot arm is approached as a learning-based visual servoing problem. Both the target image and the current image are provided, and the current arm-gripper control signal is obtained directly, without the need for prior knowledge of hand-eye calibration and depth information. The natural collaboration problem between the arm and gripper is formulated as a mapping policy function f between the images and control signals of the manipulator and gripper. This mapping function will be represented by the deep network described in Section 4.3 to achieve an adaptive mapping of feature information in image space and robot control signals.

$$[v_{arm}, w_{gripper}] = f(i_{current}, i_{target}) \tag{1}$$

As Equation 1 shows, $i_{current}$ and i_{target} represent the current and goal simplified image, the control signal v_{arm} represents the control velocity of the manipulator, and the control signal $w_{gripper}$ is the gripper width. The learned policy directly outputs manipulator and gripper control signals, thus having collaborative reaching and grasping control actions. During reaching and grasping, it can handle dynamic

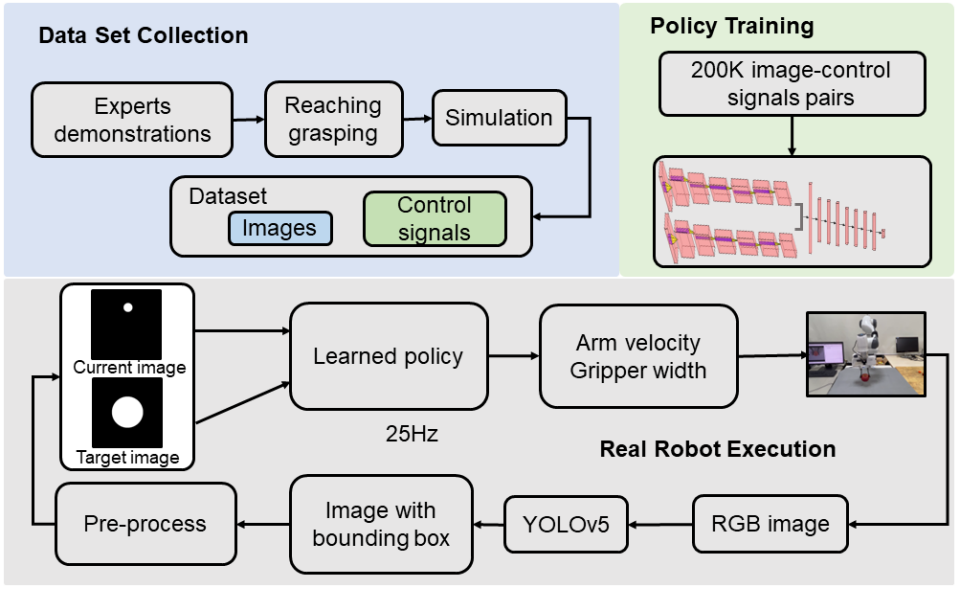


Fig. 2. The proposed framework includes three parts: data set collection, policy training, and real robot execution.

target objects and perform reactive, human-like adaptive actions, although the data set we collected in the simulation does not specifically include dynamic data.

4. Method

4.1. *Arm-gripper synergy grasping framework overview*

The Figure 2 presents the whole systematic working framework of this work. It contains three modules, including data set collection, strategy offline training, and real robot execution parts. In the data collection phase, the approach outlined in Section 4.2 is employed to execute control of the robot arm through an expert demonstration strategy. Approximately two hundred thousand pairs of images and control signals are gathered during this procedure. Regarding the policy training part, since our policy has two inputs, a parallel network is well-suited to extract our policy. The learned visual servoing policy can then be directly applied to a real robot for execution. In this process, object segmentation frames are initially acquired through off-the-shelf vision detection techniques, and subsequent image pre-processing is carried out to transform them into the iconic circular feature images collected during simulation. Finally, the current image and the target image are fed into the learned strategy to obtain the velocity control signal and the gripper control width in cartesian space of the robot arm and send them to the robot for execution. The control frequency of the whole system is set to 25 Hz.

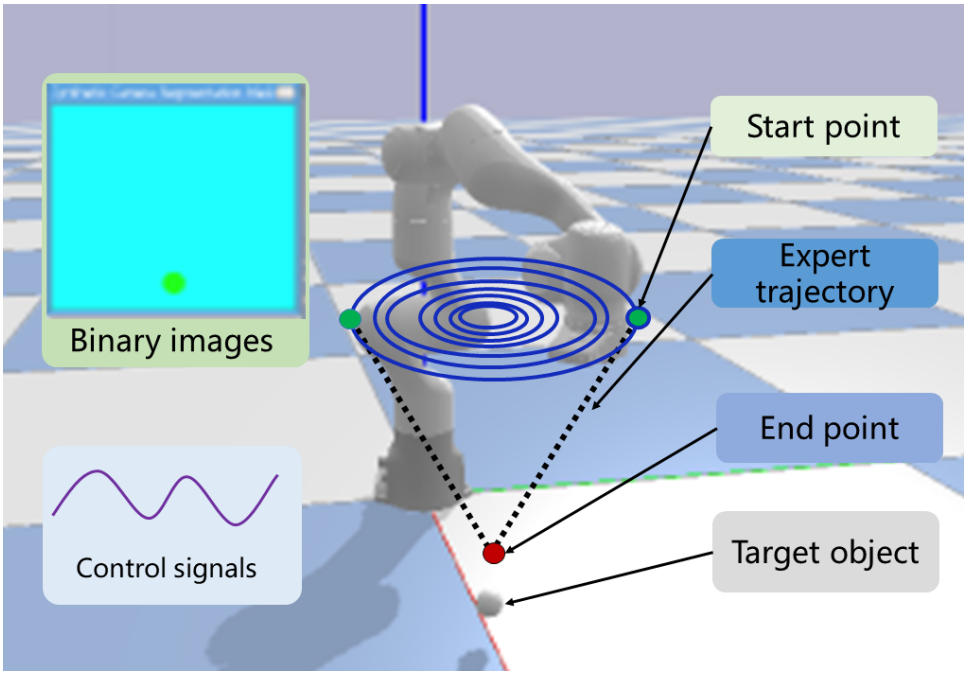


Fig. 3. The simulation environment for data set collection.

4.2. *Expert demonstration dataset collection*

This section presents how to collect the large-scale dataset in simulation automatically, which is used to train the networks to extract control policy. Dataset is the crucial factor for obtaining the correct control policy in this work. The data is automatically collected according to the designed expert reaching and grasping synergy control policy in the Pybullet simulation platform.

4.2.1. *The simulation platform for data collection*

Data is collected in the simulation while training directly on the robot, which is of key components of this work. This section introduces the simulation platform as Figure 3 shows: The target object is uniformly simplified into an iconic ball, as observed in the simulation. By applying expert skills designed in Equation 2 and Equation 3, starting from a point within the 'annual cycle of the tree,' the robot is guided to a specific target point precisely, facilitating the grasping of the target ball by our gripper. The simulation environment can be seen in Figure 3 In order to prevent collisions between the gripper and the target object, the gripper is not simulated in the simulation. Instead, the virtual gripper action is executed as designed in Equation3.

4.2.2. *Expert skills for reaching and grasping synergy*

In this work, our goal is to implement professional collaborative control strategies, which means that the dataset should be constructed under the guidance of professional robot control methods. Trajectory planning is a well-established research area. Based on robot trajectory planning techniques and our designed hand grasp merging strategy, the trajectory designed by Equation 3 is employed in the reaching and grasping process. Using a quadratic interpolation curve at the position level teaches the robot to have a smooth position and velocity control curve from the beginning to the end, maintaining position, and velocity continuity throughout the process.

$$p_i = a * t_i^2 + b * t_i + c \tag{2}$$

$$w_i = (w_{max} - w_{target}) * \frac{t_i}{t_e - t_s} \tag{3}$$

where p_i represents the 3D-space position at the t_i timestamp, a, b, c are the coefficients for the quadratic spline curve. w_i is gripper width at t_i timestamp, and w_{max},w_{target} are the maximum and target width of the gripper separately. Expert control policy can be learned from the dataset collected in this manner only if the robot is taught expert-level control skills in the simulation.

4.2.3. *Dateset composition*

Complete progress of moving the manipulator to the target position with a human-like gripper reaction is treated as a demonstration or trajectory in the simulation. In this study, the x, y, and z velocities of the manipulator in Cartesian space, along with the gripper width, are recorded as the label-control signal pairs for the datasets. At the same time, the corresponding images are matched and saved as the image pairs in the data sets. For velocity and gripper width data balance, all data are normalized to 0-1.

4.3. *Policy network and learning*

4.3.1. *Network strucure*

The deep network structure is shown in Figure 4. Our control policy is represented by a deep neural network, aiming to achieve end-to-end processing from images to control signals. The network takes as input the current camera-captured image and the target image, producing as output control signals for the robotic arm, including control velocities in the x, y, and z directions, as well as gripper width. Given the relatively simple nature of the image features used in our method, we opt for a readily available, parameter-efficient, and real-time capable AlexNet for image feature extraction. The transformation and connection from image features to control signals are accomplished using fully connected layers. Our policy needs

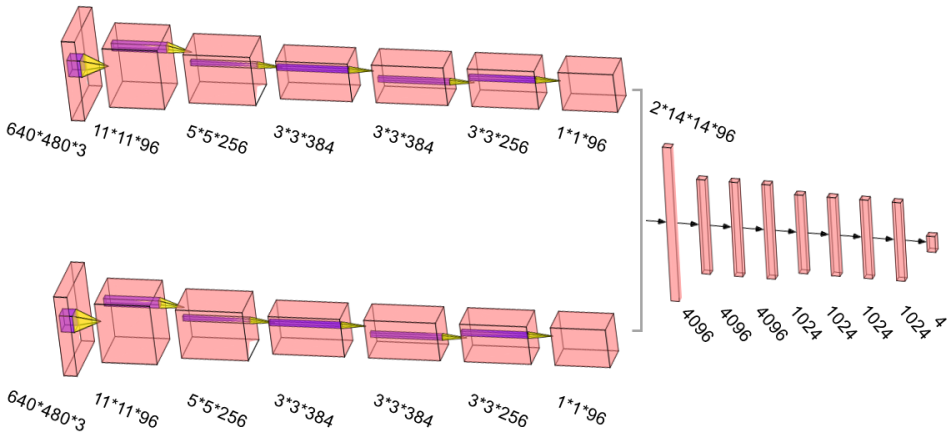


Fig. 4. The network structure. An AlexNet is employed for image feature extraction, with one branch processing the current image and another handling the target image. The features from these two branches are fused together through fully connected layers, followed by another fully connected network, which produces the final 4-dimensional control signal. This signal includes control velocities for the three dimensions of the arm and the gripper width.

to take current and goal images as the input of the network at the same time, so a siamese network[30] is intuitive to be used in our work. The siamese network has been proven to have the ability to yield accurate pose transformation between two image features. Drawing inspiration from this, the siamese network is employed for feature extraction from image pairs, with the intention of fusing them together to generate real-time direct control signals at each control step, as opposed to the output of poses between two images.

As Figure 4 shown, each branch of the network firstly makes use of the classical Alexnet network structure [31] as the encoder of each image. The pre-trained parameters of Alexnet are leveraged to enhance training efficiency, and the branch features are concatenated, followed by 6 fully connected layers to further map the control policy from image pairs. The output of the network is $O_{net} = [v_x, v_y, v_z, w_{gripper}]$, which can be directly used for real robot execution without any extra mathematical calculations and transformations.

4.3.2. *Loss*

As for the loss function, the MSE loss function in the pytorch is adopted in this work. As described in Section 4.2, each image pair corresponds to a control signal label $vec_{control} = [v_x, v_y, v_z, width]$. The output control signals refers to $\hat{v}ec_{control} = [v_x, v_y, v_z, width]$, then the loss function expression is described as follow,

$$Loss = \frac{1}{n}\sum_{i=0}^{n}\frac{1}{N}\sum_{j=0}^{N}(vec_i - \hat{v}ec_i)^2 \tag{4}$$

where N is the number of samples, n is the length of $vec_{control}$, and vec_i and $\hat{v}ec_i$ refer to the item of $vec_{control}$.

4.3.3. *Policy learning*

In this work, the learning rate is configured at 0.0001 and is scheduled to decrease by half at regular intervals during the training process. A batch size of 256 is employed, and the training utilizes the Adam optimizer. Approximately 200 thousand image pairs are employed in our training dataset. The training was conducted on a personal computer equipped with three GTX-1080Ti GPUs, operating in parallel.

4.4. *Precision and synergy metrics*

The target pose is defined as the pose when the current camera image is the same as the target image. In theory, the control velocity $v_t = [v_x, v_y, v_z]$ at the target pose is zero, and the gripper width w_t is manually setting value based on the target objects size. However, they might not be the perfect value because of policy learning and robot hardware limits, and other various system errors. To assess precision, values γ and e_w are introduced for quantifying reaching performance and gripper precision, as shown in Equation 5 and Equation 6, where w_t represents the gripper width at the target pose, and w_s denotes the predefined target object setting value.

$$\gamma = \sqrt{v_x^2 + v_y^2 + v_z^2} \tag{5}$$

$$e_w = |w_t - w_s| \tag{6}$$

$$s = \frac{v_g}{v_a} \tag{7}$$

The synergy metrics also is a novel and crucial metric in evaluating the human-like grasping in this work. It is defined as shown in Equation 7, where v_g represents the normalized closing velocity of the gripper, while v_a signifies the normalized velocity of the manipulator in Cartesian space.

5. Experimental Results and Analysis

5.1. *Experiments setup*

Validation experiments were conducted in a real-world setting, utilizing the Franka Emily 7 DOF robot. Calculations were performed on a PC equipped with a GTX-1080Ti GPU. RGB images, with a resolution of 640x480, were resized to 480x480 before being input into the learned policy. Realsense cameras were used for image

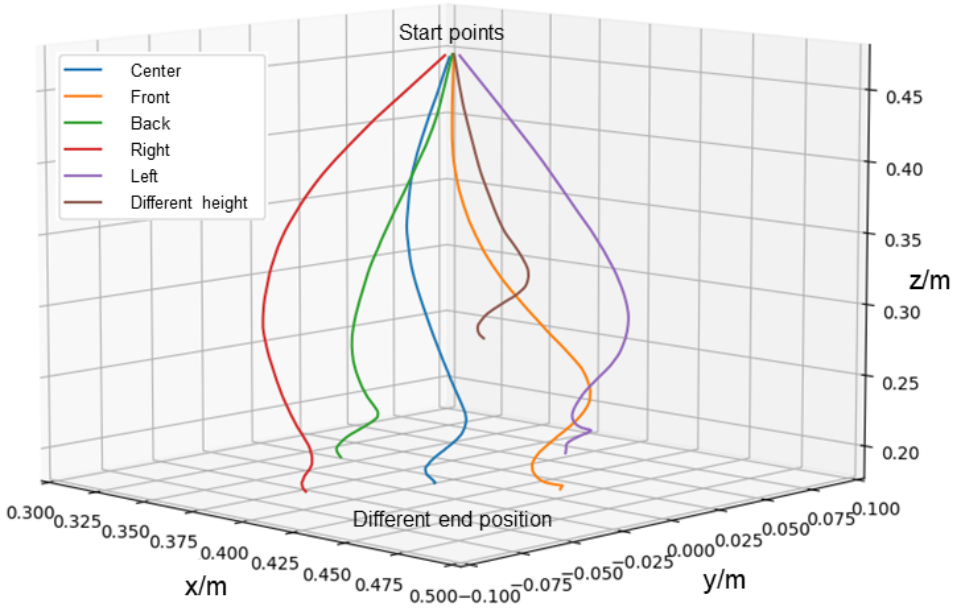


Fig. 5. The cartesian trajectories of the learn policy to grasp a static object in different positions and heights.

acquisition, though only the RGB images were employed, omitting depth information. Additionally, a low-cost web camera was utilized to illustrate the policy's effectiveness.

Subsequent experiments showcased the policy's capabilities in grasping static, dynamic, and a wide variety of objects, all while maintaining a frequency of 25Hz. It is important to emphasize that this policy does not fully address grasping planning. Rather, its function is to position the target object at the center of the gripper, allowing it to autonomously grasp center-symmetric objects exclusively.

5.2. *Arm-gripper Synergy Behavior in Grasping the Static Objects*

To evaluate the static grasping performance of our policy, two experimental scenarios were established: one involved the traditional up-down grasping, representative of the classic configuration in most bin-picking grasping algorithms, and the other entailed non-up-down grasping, where the manipulator remained stationary vertically, a feat achieved by only a limited number of data-driven bin-picking methods.

In the context of up-down grasping, the capacity to grasp a static object at various positions and heights in 3D Cartesian space is demonstrated, as depicted in Figure 5. The non-up-down grasping capability is exemplified by the retrieval of an apple model held by a human hand in a non-up-down operational environment.

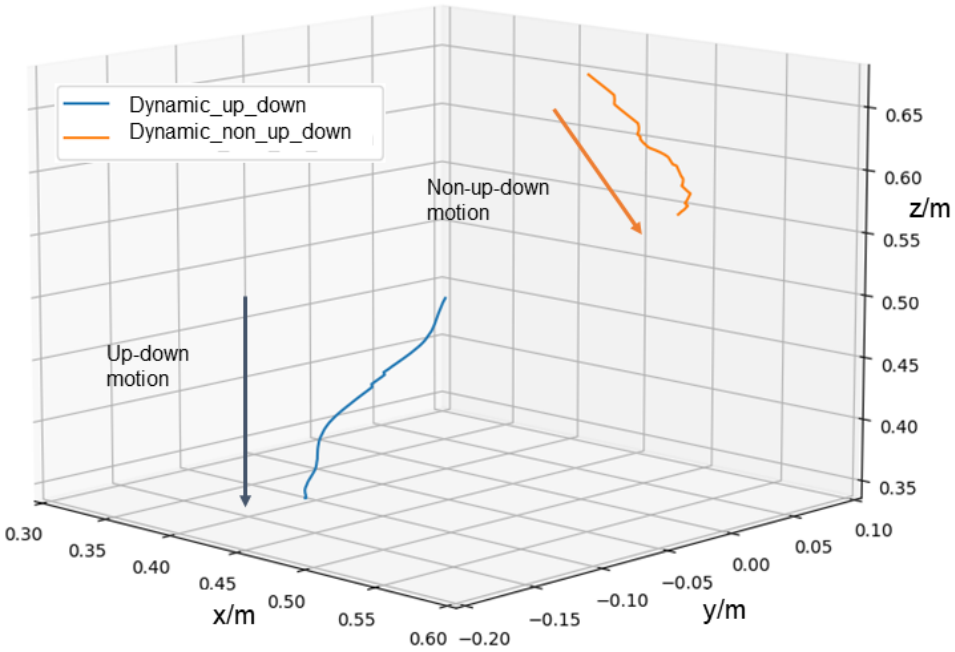


Fig. 6. The cartesian trajectories of the learn policy to grasp a dynamic object held by a human hand.

5.2.1. *Up-down grasping*

As can be seen in Figure 7, the control velocity decrease to zero with close to the target object. The closer to the target object, the smaller the velocity. All of the velocity has a quite smooth curve which is good for the manipulation control. As for the gripper width, our method produces natural, human-like actions, that is, the closer to the target object, the smaller the gripper width. The velocity in 3D-space should be zero in theory, however, it has errors because of policy precision, see Section 4.4.

Figure 5 shows the cartesian trajectories of the end effector, reflecting the learned policy having the grasping ability when the target object is located in different 3D positions.

5.2.2. *Non Up-down grasping*

The learned policy has the ability to perform grasping in a non-up-down pose, indicating its proficiency in grasping objects in 3D space. This ability can be seen in Figure 6, showing the grasping trajectory of the up-down and no-up-down settings together.

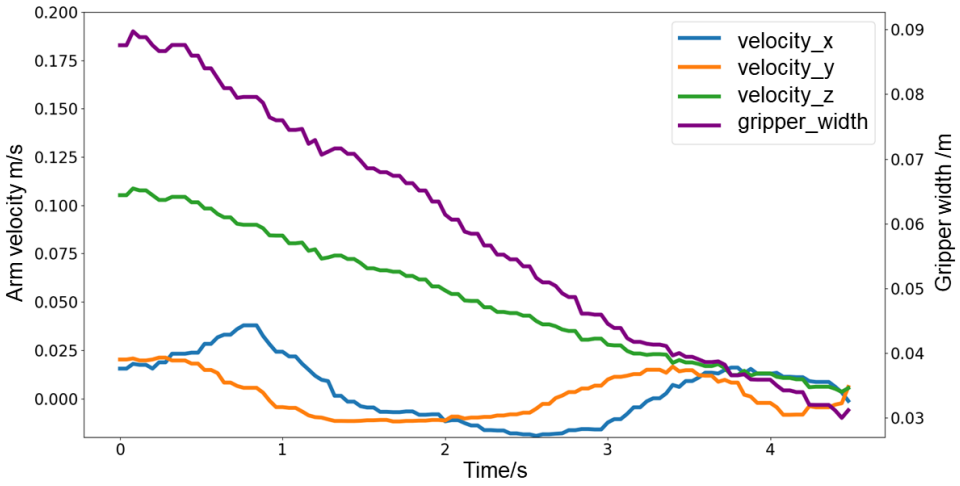


Fig. 7. The control signals of the learn policy to grasp a static object.

5.3. *Arm-gripper synergy behavior in grasping the dynamic objects*

In this section, the ability to grasp dynamic objects is exhibited. The Franka robot can chase the target objects dynamically which are taken by a human hand.

Figure 8 shows the control signals obtained from our policy. The dynamic target object induces variations in the gripper's width and the arm's velocity. The gripper's width undergoes continuous adjustments in response to changes in the target object's distance and proximity, rather than executing a simple open-close command. The arm's speed, in turn, adapts to the target object's motion: the control speed in the z-direction is directly proportional to the object's distance, while the speed in the x and y-directions ensures that the target object remains centered within the image, allowing the gripper to naturally and dynamically track the target object.

5.4. *Grasping objects with different camera FOVs*

Since the target image is fixed, different camera FOV can influence the final distance between the camera and the object. Although our specialized L-shape gripper can be adjusted in length, which can enable grasping different size objects, the size is still limited. Different FOVs can widen the range of sizes of objects that can be grasped. Hence, to achieve grasping objects with different sizes, object grasping is demonstrated using two cameras: one is a RealSense camera with a 70-degree field of view, and the other is a low-cost USB camera with a 135-degree field of view. Figure 10 showcases the grasping process with these two cameras.

Figure 10 illustrates that a larger field of view (FOV) in the camera enables the grasping of larger objects. The use of various camera FOVs demonstrates that a diverse array of objects, as depicted in Figure 9, can be grasped by our method.

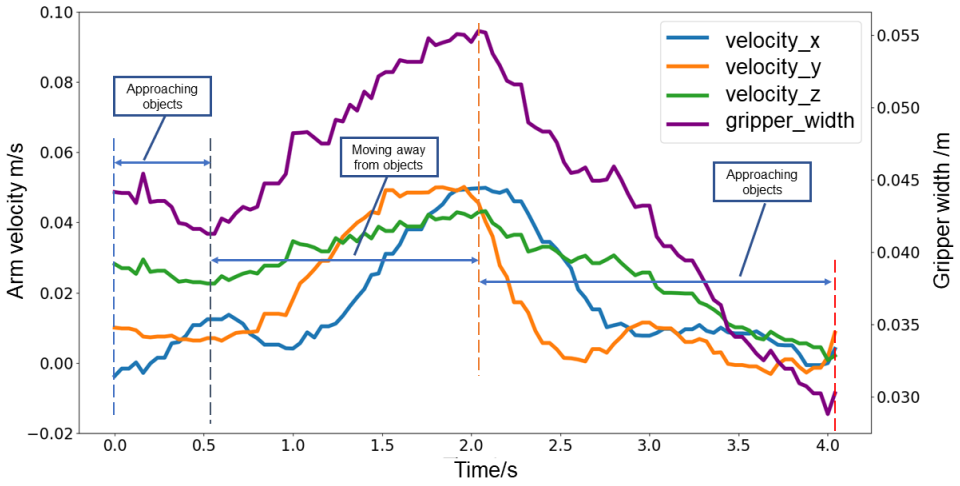


Fig. 8. The control signals of the learn policy to grasp a dynamic object held by a human hand.

Fig. 9. The experiment objects, the objects within the outlined boxes in the picture are the model, while the others are real objects.

5.5. *Precision and synergy performance*

In the grasping task, the control signals, including velocity, are not as accurate as their ideal value of zero. As a result, the velocity is set to zero when it reaches the threshold value, which corresponds to a 1mm position increment within a single

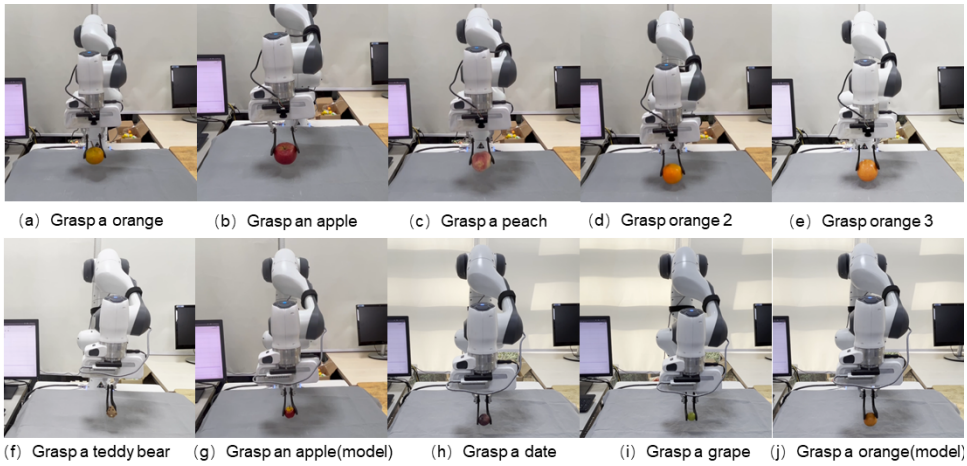


Fig. 10. The proposed framework can grasp a wide range of objects. (a)-(e) using a low-cost USB camera; (f)-(j) using RGB output of Realsense camera. (Partial display, see the accompanying video for all.)

Table 1. Visual servoing precision of two different target objects.

No. of Test	Apple/γ (m/s)	Apple/e_w	Orange/γ (m/s)	Orange/e_w
Test 1	0.01005	-0.01582	0.00262	-0.01461
Test 2	0.01005	0.0009	0.00352	-0.02969
Test 3	0.00306	-0.02940	0.00355	-0.02053
Test 4	0.01042	-0.01377	0.00328	-0.02193
Test 5	0.01019	0.00565	0.00251	-0.01580
Test 6	0.01205	0.00122	0.00262	-0.01461
Test 7	0.00905	-0.00480	0.00271	-0.02317
Test 8	0.01256	0.01627	0.02048	0.04683
Test 9	0.01180	-0.01768	0.00104	-0.02083
Test 10	0.00822	-0.00677	0.00267	-0.01517
Average	**0.00975**	**-0.00641**	**0.00451**	**-0.01295**

control period. Subsequently, the gripper is closed simultaneously. Nevertheless, it should be noted that this action does not represent the actual limit or precision of the visual servoing control.

To evaluate the true precision based on Equation 5 and Equation 6, precision test experiments were conducted, in which the threshold value was set to zero, as prescribed by theory.

During the test experiments, the gripper extension was deactivated to prevent

contact with the target object. Only the gripper width, as per the policy output, was recorded without execution, and the control portion of the robot arm was executed until the arm reached a complete standstill. In order to mitigate the influence of single target object specificity, two representative objects, namely an apple model and an actual orange, were selected as target objects. Each of these objects underwent a series of 10 repeated experiments.

The result is shown in Table 1. It is clear to see it has quite a high-velocity control precision. The average control precision of velocity is 9.75mm/s and 4.51mm/s for apple and orange separately under a control frequency of 25Hz. As for the gripper width precision, the data in Tbale 1 is normalized to 0-1, which is 0.5128mm and 1.036mm for the Franka gripper.

The defined arm-gripper synergy metric Equation 7 is presented in Figure 11, comparing with the traditional method: Approach the object first and then perform the grasp. It can be seen that our proposed synergy metric represents the synergy of the arm-gripper properly: The closer to the target object, the bigger the degree of synergy. This strategy design for arm-gripper synergy is consistent with the intuition of grasping that not much synergy is required when moving away from the object and bigger synergy is required when approaching the target object.

5.6. *Generalization test*

In this work, the control signals of our strategy are limited to positions in 3D space without orientation, and the learned policy is designed to master the case where the plane where the target object is located is parallel to the horizontal plane of the manipulator. However, in order to test and gain a deeper understanding of the learned strategy, generalization test experiments were conducted to complement it. In the default setting of our grasping task, the horizontal plane where the camera is located and the plane where the target object is located is parallel. In the generalization test experiments, slight adjustments were made to both angles to observe how the learned strategy would handle this situation. In the attached video, it is evident that the learned strategy continues to achieve successful grasps at small tilt angles, such as 30 degrees. However, when the angle exceeds 45 degrees, grasp failures occur. This is attributed to the non-parallel orientation of the camera plane and the plane where the target object is situated, necessitating varying proportions of x, y, and z velocities; otherwise, failure occurs due to disparate velocity. Nevertheless, our proposed method demonstrates a degree of generality concerning the angle between the camera plane and the target object plane.

5.7. *Comparisons*

In order to emphasize the performance of our approach, a comparison with various state-of-the-art grasping methods is conducted across multiple dimensions, as presented in Table 2. One of the noteworthy features of our method is its demonstrated capability for adaptive arm-gripper synergy in both static and dynamic grasping

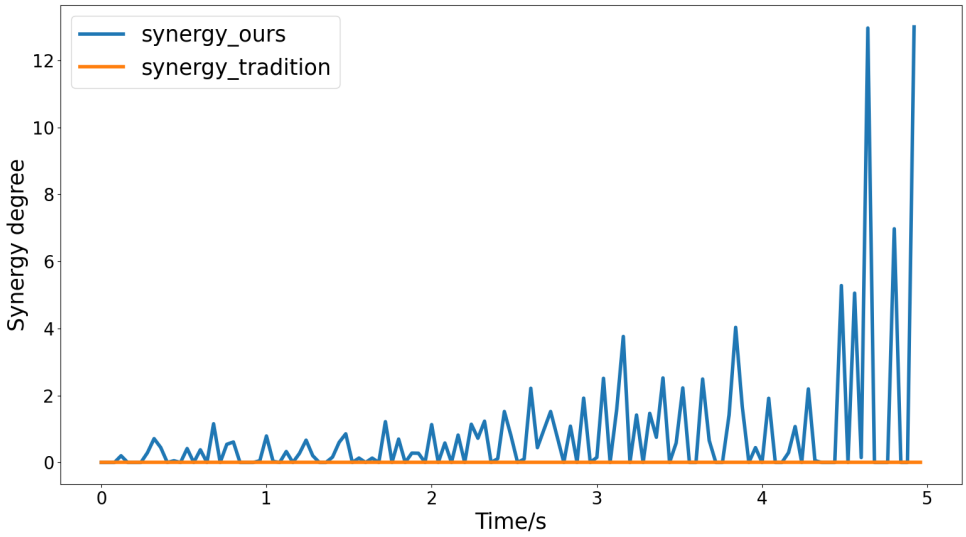


Fig. 11. The arm-gripper synergy performance of grasping an orange.

Table 2. Comparisons with other works in proposed items.

	[11]	[28]	[9]	[10]	Ours
Closed-loop	✓	✓	×	×	✓
Arm-gripper synergy	×	×	×	×	✓
Real-time	✓	✓	×	×	✓
No calibration required	×	✓	×	×	✓
No depth info required	×	✓	×	×	✓

scenarios, as illustrated in Figure 11. In contrast, other studies do not exhibit this capacity.

Furthermore, our method does not require precise camera calibration or depth information during the grasping process. This is a significant advantage over some other methods, such as baseline method[11], which stops updating the real-time grasping pose when the distance between the hand-eye camera and the target object is less than 15cm. As a result, the method is likely to fail when the target object moves within this range. Our method does not suffer from this limitation, making it more robust and reliable in real-world grasping scenarios.

Note that work[28] is a similar work to ours in that it also achieves continuous servo grasping without requiring precise camera calibration. However, unlike our method, it does not perform arm-gripper synergy in the grasping process.

As shown in Table 2, our method outperforms other state-of-the-art grasping

methods in all five aspects of the grasping process. In contrast, some of the other methods lack one or more of these abilities. Notably, our method's adaptive arm-gripper synergy ability is unique among the compared methods. This feature allows our method to yield adaptive gripper actions shown in Figure 8, resulting in more natural grasping performance similar to that of humans.

6. Limitation

Our approach enables dynamic and collaborative arm grasping in real-time. However, it is important to acknowledge certain limitations that warrant consideration. One limitation stems from the design of the camera integrated into the gripper. In some instances, the gripper's placement between the target object and the optical center of the camera can obstruct the camera's view, potentially leading to grasp failures. Furthermore, our current control signal is primarily position-based and does not account for orientation, restricting autonomous grasping to centrally symmetrical objects. Lastly, while our method facilitates the grasping of various centrosymmetric objects of different sizes, it necessitates the use of multiple cameras with varying fields of view. These limitations underscore the areas of our focus for future research.

7. Conclusion

In this study, a novel framework for arm-gripper synergy control through the acquisition of a visual servoing controller is presented. It is demonstrated that the proposed approach allows for the attainment and grasping of multiple target objects with a natural, adaptable, and collaborative behavior exhibited between the manipulator and gripper. The learned visual servoing controller facilitates the realization of dynamic arm-gripper collaborative grasping, even in the absence of precise calibration and depth information.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (U2013602, 52075115, 51521003, 61911530250), National Key R&D Program of China(2020YFB13134), Self-Planned Task (SKLRS202001B, SKLRS202110B) of State Key Laboratory of Robotics and System(HIT), Shenzhen Science and Technology Research and Development Foundation (JCYJ20190813171009236), Basic Research on Free Exploration of Shenzhen Virtual University Park (2021Szvup085) and Basic Scientific Research of Technology (JCKY2020603C009). Qiang Li is supported by the "DEXMAN" project funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation)-project number(410916101)

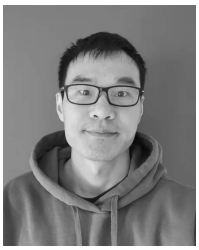
References

1. Maskrcnn K. He, G. Gkioxari, P. Dollar, and R. Girshick, Mask r-cnn, in Proceedings of the *IEEE International Conference on Computer Vision* (ICCV, 2017), pp. 2980-2988.

2. Bohg J, Barck-Holst C, Huebner K, et al. Towards grasp-oriented visual perception for humanoid robots[J]. International Journal of Humanoid Robotics, 2009, 6(03): 387-434.
3. A. Provenzale, F. Cordella, L. Zollo, A. Davalli, R. Sacchetti, and E. Guglielmelli, A grasp synthesis algorithm based on postural synergies for an anthropomorphic arm-hand robotic system, Proceedings of the *IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics*(IEEE, 2014), pp. 958–963.
4. J. Bohg, A. Morales, T. Asfour, and D. Kragic, Data-driven grasp synthesis-a survey, *IEEE Transactions on Robotics* **30**(304) (2014) 289–309.
5. S. Wang, W. Hu, L. Sun, X. Wang, and Z. Li, Learning adaptive grasping from human demonstrations, *IEEE/ASME Transactions on Mechatronics* **10** (2022) 3865-3873.
6. Kleeberger, Kilian, Richard Bormann, Werner Kraus, and Marco F. Huber. A survey on learning-based robotic grasping. *Current Robotics Reports* **1** (2020): 239-249
7. Chen, Pengzhan, and Weiqing Lu. Deep reinforcement learning based moving object grasping. *Information Sciences* **565** (2021) 62-76.
8. Brock O, Fagg A, Grupen R, et al. A framework for learning and control in intelligent humanoid robots[J]. International Journal of Humanoid Robotics, 2005, 2(03): 301-336.
9. J. Mahler, M. Matl, V. Satish, M. Danielczuk, B. DeRose, S. McKinley, and K. Goldberg, Learning ambidextrous robot grasping policies, *Science Robotics*, **4**(26) (2019), p.eaau4984
10. Fang, Hao-Shu, Chenxi Wang, Minghao Gou, and Cewu Lu. Graspnet-1billion: A large-scale benchmark for general object grasping. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR, 2020), pp. 11444-11453.
11. Morrison, Douglas, Peter Corke, and Jürgen Leitner. Learning robust, real-time, reactive robotic grasping. *The International journal of robotics research* **39**(2-3) (2020) 183-201.
12. A. A. Shahid, L. Roveda, D. Piga, and F. Braghin, Learning continuous control actions for robotic grasping with reinforcement learning, Conference Proceedings - *IEEE International Conference on Systems, Man and Cybernetics* (IEEE, 2020), pp. 4066–4072.
13. Y. Y. Tsai, H. Xu, Z. Ding, C. Zhang, E. Johns, and B. Huang, Droid: Minimizing the reality gap using single-shot human demonstration, *IEEE Robotics and Automation Letters*, **6**(4) (2021), 3168–3175.
14. Rosell J, Suárez R, García N, et al. Planning grasping motions for humanoid robots[J]. International Journal of Humanoid Robotics, 2019, 16(06): 1950041.
15. Marturi, Naresh, Marek Kopicki, Alireza Rastegarpanah, Vijaykumar Rajasekaran, Maxime Adjigble, Rustam Stolkin, Aleš Leonardis, and Yasemin Bekiroglu. Dynamic grasp and trajectory planning for moving objects. *Autonomous Robots* **43** (2019): 1241-1256.
16. W. Hu, C. Yang, K. Yuan, and Z. Li, Learning Motor Skills of Reactive Reaching and Grasping of Objects, *IEEE International Conference on Robotics and Biomimetics* (ROBIO) (IEEE, 2021), pp. 452–457.
17. A. Depierre, E. Dellandrea, and L. Chen, Jacquard: A large scale dataset for robotic grasp detection, *IEEE International Conference on Intelligent Robots and Systems* (IEEE, 2018), pp. 3511–3516.
18. Z. Zhang, C. Zhou, Y. Koike, and J. Li, Single RGB Image 6D Object Grasping System Using Pixel-Wise Voting Network, *Micromachines* bf 13(2) (2022), 293,
19. S. Peng, Y. Liu, Q. Huang, H. Bao, and X. Zhou, PVNet: Pixel-wise Voting Network for 6DoF Pose Estimation. In Proceedings of the *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2019), pp. 4561-4570.
20. W. Kehl, F. Manhardt, F. Tombari, S. Ilic, and N. Navab, "SSD-6D: Making RGB-Based 3D Detection and 6D Pose Estimation Great Again." [Online]. Available:

https://wadimkehl.github.io/

21. S. Ainetter and F. Fraundorfer, End-to-end trainable deep neural network for robotic grasp detection and semantic segmentation from rgb, Proceedings -*IEEE International Conference on Robotics and Automation* (IEEE, 2021), pp. 452–13.
22. V. Lepetit, F. Moreno-Noguer, and P. Fua, Epnp: An accurate o(n) solution to the pnp problem, *International Journal of Computer Vision*, **81**(2) (2009), 55–166.
23. D. Kragic and H. I. Christensen, Survey on Visual Servoing for Manipulation. *Computational Vision and Active Perception Laboratory, Fiskartorpsv* **15** (2002).
24. P. I. Corke, D. Morrison, P. Corke, and J. Rgen Leitner, Robust Robotic Manipulation View project Learning to Act on Multi-Modal Data (LAMDa) View project Learning robust, real-time, reactive robotic grasping, *Article in The International Journal of Robotics Research*, 2019.
25. Hutchinson, Seth, and F. Chaumette. Visual servo control, part i: Basic approaches. *IEEE Robotics and Automation Magazine* **13**(4) (2006) 82-90.
26. D. Kappler, F. Meier, J. Issac, J. Mainprice, C. G. Cifuentes, M. Wuthrich, V. Berenz, S. Schaal, N. Ratliff, and J. Bohg, Realtime perception meets reactive motion generation, *IEEE Robotics and Automation Letters* **3**(3) (2018) 1864–1871.
27. Lenz, Ian, Honglak Lee, and Ashutosh Saxena. Deep learning for detecting robotic grasps. *The International Journal of Robotics Research* **34**(4-5) (2015): 705-724.
28. S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection, T*he International journal of robotics research* bf 37(4-5) (2018) 421–436.
29. Valassakis, Eugene, Georgios Papagiannis, Norman Di Palo, and Edward Johns. Demonstrate Once, Imitate Immediately (DOME): Learning Visual Servoing for One-Shot Imitation Learning. In 2022 *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2022), pp. 8614-8621.
30. C. Yu, Z. Cai, H. Pham, and Q. C. Pham, Siamese convolutional neural network for sub-millimeter-accurate camera pose estimation and visual servoing, *IEEE International Conference on Intelligent Robots and Systems* (IEEE, 2019), pp. 935–941.
31. A. Krizhevsky, I. Sutskever, and G. E. Hinton, Imagenet classification with deep convolutional neural networks, *Communications of the ACM***60**(5) (2017) 84–90.



Shuaijun Wang received the M.S. degree in Mechatronics Engineering from Harbin Institute of Technology in 2015. He is currently working toward his Ph.D. degree in the State Key Laboratory of Robotics and System, Harbin Institute of Technology (HIT), Harbin, China, and also was a visiting researcher in the Advanced Intelligent Robotics (AIR) Lab, School of Informatics, University of Edinburgh, UK in 2019-2021. During 2022.12-2023.06, he worked as a researcher in Tencent Robotics X lab. His research interests include robotic grasping, manipulation and embodied AI.

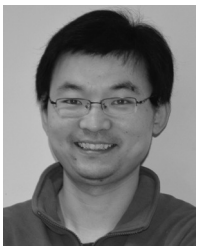
Lining Sun received his Ph.D. degree in Engineering from the Mechanical Engineering Department of Harbin Institute of Technology in 1993. He is a National Outstanding Youth Fund Winner, Changjiang Scholar Distinguished Professor by the Ministry of Education. He formed an internationally competitive robotics team and made outstanding contributions to the creation and development of robot-related disciplines in China.

Mantian Li received the B.E. and M.S. degrees in vehicle engineering and the Ph.D. degree in mechatronics engineering from the Harbin Institute of Technology, Harbin, Heilongjiang, China, in 1998, 2000, and 2006, respectively. His research interests include quadruped robot and industrial robot technology.

Pengfei Wang received the B.E., M.E., and Ph.D. degrees in mechatronics engineering from the Harbin Institute of Technology, Harbin, China. He has been an Associate Professor with the Harbin Institute of Technology. His current research interests include robust control problems and advanced robot control theory.

Fusheng Zha received the Associate Degree in mechatronics engineering from Wuhu Radio and TV University, in 1997, the M.S. degree in mechanical design and theory from the Lanzhou University of Technology, in 2005, and the Ph.D. degree in mechatronics engineering from the Harbin Institute of Technology, in 2012. His research interests include CPG control, quadruped robot, and neural networks.

Wei Guo received the B.E., M.E., and Ph.D. degrees in mechatronics engineering from the Harbin Institute of Technology, Harbin, China. She has been a Professor with the Harbin Institute of Technology. Her current research interests include control system design and advanced robot control theory.



Qiang Li Qiang Li received the Ph.D. degree in pattern recognition and intelligence systems from the Shenyang Institute of Automation (SIA), Chinese Academy of Sciences (CAS), Shenyang, China, in 2010. He was awarded stipend from the Honda Research Institute Europe, Offenbach, Germany. From 2009 to 2012, he started his postdoctoral research with the CoR-Lab, Bielefeld University, Bielefeld, Germany. He is currently a Principal Investigator of "DEXMAN" sponsored by the Deutsche Forschungsgemeinschaft (DFG), Bonn, Germany, and is working with Neuroinformatics Group, Bielefeld University. His current research interests include visuo/tactile servoing and recognition, sensory-based robotic dexterous manipulation, and robotic calibration and dynamic control.