Robotic perception and grasp in unstructured environments

Dong, Huixu

2019

Dong, H. (2019). Robotic perception and grasp in unstructured environments. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/89885>

<https://doi.org/10.32657/10220/47731>

ROBOTIC PERCEPTION AND GRASP IN UNSTRUCTURED ENVIRONMENTS

HUIXU DONG

**SCHOOL OF MECHANICAL & AEROSPACE ENGINEERING
NANYANG TECHNOLOGICAL UNIVERSITY**

A thesis submitted to Nanyang Technological University
in partial fulfillment of the requirements for the
Degree of Doctor of Philosophy

2019

ROB. PER. AND GRA. IN UNSTR. ENV. H. DONG

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

02/25/2019

Dong Huixu

.....

Date

.....

Huixu Dong

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

02/25/2019

.....

Date

.....

I-Ming Chen

Authorship Attribution Statement

This thesis contains material from 8 paper(s) published in the following peer-reviewed journal(s) / from papers accepted at conferences in which I am listed as an author.

Chapter 3 is published as H. Dong, E. Asadi, G. Sun, D. K. Prasad, I-Ming Chen, “Real-time Robotic Manipulation of Cylindrical Objects in Dynamic Scenarios through Elliptic Shape Primitives,” IEEE Transactions on Robotics (2018); H. Dong, I-Ming Chen, and D. K. Prasad, “Robust Ellipse Detection via Duality Principle with a False Determination Control (ICIP), Beijing, China, 17-20 September, 2017.

The contributions of the co-authors are as follows:

- Prof. Chen provided the initial project direction.
- I prepared the manuscript drafts. The manuscript was revised by Dr Dilip.
- Dr Ehsan and Dr.Sun assisted in performing the experiments.

Chapter 4 is published as H. Dong, D. K. Prasad, Q.Yuan, J.Zhou, E.Asadi, I-Ming Chen. " Efficient Pose Estimation from Single RGB-D Image via Hough Forest with Auto-Context," IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS), October,1-5, 2018, Madrid, Spain.

The contributions of the co-authors are as follows:

- Prof. Chen provided the initial project direction.
- I prepared the manuscript drafts. The manuscript was revised by Dr Dilip.
- Dr Yuan, Jiadong Zhou and Dr.Ehsan assisted in performing the experiments.

Chapter 5 is published as H. Dong, G.Sun, W.Pang, E.Asadi, D.k. Prasad, I-Ming Chen." Fast Ellipse Detection via Gradient Information for Robotic Manipulation of Cylindrical Objects." IEEE Robotics and Automation Letters (2018); H. Dong, D. K. Prasad, and I-Ming Chen, “Accurate detection of ellipses with false detection control at video rates using a gradient analysis,” Pattern Recognition, 2018; H. Dong, E.

Asadi, G. Sun, D. K. Prasad, I-Ming Chen, "Real-time Robotic Manipulation of Cylindrical Objects in Dynamic Scenarios through Elliptic Shape Primitives," IEEE Transactions on Robotics (2018).

The contributions of the co-authors are as follows:

- Prof. Chen provided the initial project direction.
- I prepared the manuscript drafts.
- The manuscript was revised by Dr Dilip, Dr.Ehsan.
- Dr Sun, W. Pang assisted in performing the experiments.

Chapter 6 is published as H. Dong, E. Asadi, C. Qiu, J. Dai, and I-Ming Chen. "Geometric design optimization of an under-actuated tendon-driven robotic gripper." Robotics and Computer-Integrated Manufacturing (2017); H. Dong, E. Asadi, C. Qiu, J. Dai, and I-Ming Chen. " Grasp analysis and optimal design of robotic fingertip for two tendon-driven fingers." Mechanism and Machine Theory (2018); H. Dong, C Qiu, DK Prasad, Y Pan, J Dai, IM Chen." Enabling grasp action: Generalized quality evaluation of grasp stability via contact stiffness from contact mechanics insight." Mechanism and Machine Theory (2018).

The contributions of the co-authors are as follows:

- Prof. Chen provided the initial project direction.
- I prepared the manuscript drafts and it was revised by Dr.Ehsan.
- Dr.Qiu and Prof.Dai provided revised comments。

02/25/2019

Dong Huixu

.....

Date

.....

Huixu Dong

Abstract

Perception and grasp are crucial capabilities for robots to perform desired services in unstructured environments. This thesis presents important insights and concepts related robotic perception and grasp in unstructured scenarios. In this thesis, three specific problems related to this topic have been studied, viz., tracking of cylindrical objects, grasping of static and dynamic cylindrical objects based on the proposed ellipse detection, pose estimation of multiple objects and occluded objects in cluttered environments and the optimal design of under-actuated robotic gripper for realizing stable grasps. For the problem of the ellipse detection, two critically important problems have been addressed. The proposed detector provides either faster or more accurate ellipse detection results than the current state-of-the-art methods, irrespective of challenging scenarios such as occluded or overlapping ellipses. For the problem of pose estimation of objects, we propose a highly efficient learning approach integrated by the contextual information to estimate pose of the textured or texture-less objects for grasping purposes in a cluttered environment where the objects might be partially occluded. It has been indicated that the proposed method is superior to several state-of-the-art works. The proposed perception algorithms impose the constraints in the scenarios where the severe occlusions result in the lack of visibility. For the problem of the optimal design of under-actuated robotic gripper, the mathematical model between the active and contact forces has been expressed and the geometric model of transmission characteristics determined by the tendon routes for reducing the resistance has been explored for determining the dimension parameters of the gripper. Practical experiments are performed by the designed gripper to validate the proposed designed approach. The utility of these algorithms has been shown using several series of robotic grasp experiments with successful rates of over 80% in various difficult scenarios, including tracking cylindrical objects, grasping static and dynamic cylindrical objects, grasping textured and texture-less by estimating poses of multiple objects and occluded objects.

Acknowledgement

I convey my deepest love to my love, my mother, my father, my cute nephew, my sister, my brother-in-law, for their continuous love, encouragements, understanding and support. In reaching this point in my life, I owe enormous gratitude to many individuals and groups; too many to mention or even count. Nevertheless, there are several people I would like to convey special thanks.

First of all, I would like to express my sincere gratitude to my advisor, Professor Chen I-Ming, for his patience and pertinent guidance during my research. I am willing to express my most sincere gratitude to him for providing me an opportunity to pursue a doctorate. His encouragement and supervision helped me in gaining the confidence in conducting my research. His visions and broader insight played an important role in the progress of this thesis. I am very grateful to Prof.Chen for giving me the independence to pursue independent research on the topic of robotic perception and grasp. He always deals with the Ph.D student's issues as aggressively as he could. He gave me some excellent suggestions on the future road when I want to continue my academic career.

Much appreciate to my mentor, Dr. Dilip Kumar Prasad, for giving me enormous and important helps and supports, regardless of either the academic or the philosophical. He guided and helped me to walk away from the most difficult moment during the period of pursuing the Ph.D degree. He is a real academician and educator. My research philosophy is trained in a very rigorous way by Dr. Dilip, which will help me in future as well. He has motivated me to keep high expectation from myself in terms of the quality of work. He always encourages me to broaden my academic insights and give me some constructive suggestions on the future road. His goodness and spirits of passions and pureness on the research make him unforgettable.

I am especially grateful to my mentor, Dr. Ehsan Asadi, for his substantial support and important guidance. His thoughts and composites in the research undisturbed from the external obstacle in research environment, which is

infrequently seen among researchers. He helped me to revise the manuscripts even at the midnights and on the weekends. I am very impressed and benefitted from his rigour in research. My research also had impacted beneficially from this rigour I learn much from his strong engineering, mathematical background and wide knowledge in the software and hardware in robotics.

I specifically owe much to my mentor, Dr. Sun Guangbin, for his abundant helps and indispensable guidance over the more than two years. He always helps me to solve many academic or engineering problems by spending his personal time. It was always a pleasure to learn from him and his strategy to solve an intractable problem whenever it became too nebulous to see clearly. He also gave many excellent and constructive suggestions and helped in mentoring my research vision and future career path.

I must provide special grateful to my mentor, Dr. Yuan Qilong, for his pivotal advice, assistance, friendship and encouragement, regardless of either research or life. Since I have moved to Nanyang Technological University to carry on my research, I have received the significant support from Dr. Yuan and providing me with his deep thoughts about the research problem. His profound insights of the life and research are impressive and rare. He has taught me to always look at problems from different perspectives.

I also got many help and encouragement from Prof. Yeo Song Huat, Prof. Lum Guozhan, Prof. Dai Jiansheng, Ching-Yen Weng. The staffs of Robotics Research Centre also provide many helps and advices. I sincerely appreciate their help and support.

Huixu Dong

List of Symbols

x_i, y_i	The x -coordinate and y -coordinate of a pixel p_i in a 2D image
η_i	The phase of the gradient
e	The arc
τ_{abc}	A set of three arcs
P_i	The endpoint of arc, $i = 1, 2, 3, 4$
C	The centre of ellipse
ρ	The orientation of an ellipse
B	The ellipse minor semi-axes length
A	The ellipse major semi-axes length
x_c, y_c	The coordinate of the centre
N_B	The number of pixels in the vector $\mathcal{B}$.
$E_{V,i}$	The i^{th} parameters in the vector V
$E_{W,i}$	The i^{th} parameters in the vector W
ε_d	A detected ellipse
ε_g	The ground-truth ellipse
Φ and ε_g	The overlap ratio that is the Jaccard index of similarity between ε_d
6D	Six dimensions
3D	Three dimensions
CNN	Convolutional Neural Network
V	A patch size
$\mathcal{P}_i$	The appearance of the local patch at the location i of the centre pixel
J_i	The extracted multiple features at a patch at the patch $\mathcal{P}_i$
c_i	An object class
d_i patch	The offset from the centroid of the object to the centre of the sample
R_1, R_2	Two rectangles within the patch boundaries
$\tilde{\theta}$	The mean of 6D pose

Γ	The full covariance matrix for each mode
TDMs	Tendon-driven mechanisms
$\mathcal{L}, \mathcal{R}$	The left and the right, respectively
Γ^α	The covariance matrix among the location vectors
Γ^β	The covariance matrix among the rotation angle vectors
$\mathcal{M}$	The set of mixture components
ϑ	The support weight
Q	The posterior probability of vote elements in given patches reaching the leaf
ζ	The adaptive weight
$\mathcal{N}_i$	The small neighbourhood around the centre pixel in $\mathcal{P}_i$
$\mu_{i,T}$	The mean value with the highest mixture weight of the pose distribution
H	The object pose
Ω	The observed image
ω	The vector of learnable parameters consists of CNN weights
$\mathbb{R}(H)$	The rendered image
n	The number of output units
SGD	Stochastic gradient descent
M	An object model
R_{gr}	The ground truth rotation
T_{gr}	The ground truth translation
R_{es}	The estimated rotation
T_{es}	The estimated translation
TDM	Tendon-driven mechanism
L	The number of tendons
N	The number of joints
A	The number of actuators
l_0	The initial extension for the passive tendon.

$\dot{\theta}$	The joint angle velocity vector
$\dot{l}$	The tendon extension rate
J_k	A Jacobian matrix mapping the angle velocity vector to the tendon extension rate.
R_a	The diagonal matrix
F_t	The vector of tendon tension force
k_i	The stiffness of linear springs for the joints
F_a	The tension force of the active tendon
$\Delta\theta_i$	The variations of joint angles with respect to the initial values
μ	The friction coefficient between a tendon and the object
$F_{external}$	The contact tangent force generated by the active tendon tension force
F_{resist}	The resisting force for keeping the current state
η	The efficiency of the force transmission for the tendon wrapping the mandrel
ε	The initial intersection angle between the two phalanges
δ_1	The total contact force
δ_2	The discrete magnitude for the contact forces
GA	Genetic algorithm
d	The width of the maximum stretching state
w_{obj}	The maximum desired width of the grasped object

Table of Contents

Chapter 1	Introduction.....	1
1.1	Motivation	1
1.2	Objective	3
1.3	Organization of Report.....	4
Chapter 2	Literature Review.....	6
2.1	Perception by 2D Information.....	7
2.2	Perception by 3D Information.....	9
2.3	Synthesis of Robotic Perception and Grasp	13
2.4	Design Optimization of Under-actuated Robotic Gripper	14
2.5	Summary	16
Chapter 3	Visual Perception based on Ellipse Detection	18
3.1	Arc Segmentation and Classification	20
3.2	Ellipse Detection	27
3.3	Threshold Confirmation and Detector Comparisons	36
3.4	Summary	46
Chapter 4	Visual Perception based on RGB-D Images	47
4.1	Methodology	49
4.2	Evaluations and Discussions of Estimation Performance	66
4.3	Summary	73

Chapter 5	Experiments and Discussions of Robotic Grasp	75
5.1	Robotic System and Platform.....	75
5.2	Manipulations of Cylindrical Objects	76
5.3	Experiments and Discussion for Novel Picking and Placing Strategy	87
5.4	Summary	91
Chapter 6	Geometric Design Optimization of Under-actuated Gripper.....	93
6.1	Desired Outcome.....	95
6.2	Problem Modelling.....	98
6.3	Design Optimization	108
6.4	Optimization Procedure and Results	112
6.5	Experimental Results.....	118
6.6	Summary	120
Chapter 7	Conclusion	121
7.1	Contributions.....	122
7.2	Future Work	127

List of Figures

Figure 2-1: Several cases of ellipse detectors proposed by Prasad et al. (A), Jia et al. (B), Liu et al. (C), Mai et al. (D).....	8
Figure 2-2: The pose estimation methods from Brachmann et al. (A), Doumanoglou et al. (B), Kehl et al. (C), Choi et al. (D).....	10
Figure 2-3: The grasp systems of Jiang et al. (A), Levine et al. (B), Lenz et al. (C), Saxena et al. (D), Dmitry et al.(E) and Jeffrey et al. (F).	14
Figure 2-4: The Velo gripper (A); the Model S hand (B); the GR2 Gripper (C); the DEXMART hand (D); the hand of the DLR Hand Arm System (E); tendon-driven index finger (F).	16
Figure 3-1: Tracking ellipses in a dynamic scenario.	19
Figure 3-2: The image by a pre-processing. The initial image (A); the edge images along two edge directions by gradient signs (B, C); the edge images after the process of 8-connectivity (D, E); the arcs enclosed by oriented bounding boxes (F, G); the image after pre-processing (H).	21
Figure 3-3: A curve fitted by a series of line segments. Line segments with arrows represents the directions of curvatures of arcs and the red rectangle frames denote bounding boxes enclosing arcs extracted.	23
Figure 3-4: Arc classification based on the convexity-concavity.	25
Figure 3-5: Arc classification.....	27
Figure 3-6: Illustration of the calculation method of the elliptic center. The methods of estimating the elliptic center proposed by others (A) and us (B, C). The red line segments are the two longest ones in all line segments(B).	29
Figure 3-7: Procedure of a center estimation by mean-shift clustering algorithm. The intersections generated by two pairs of arcs (A); the center	

clusters enclosed by the dash circles with the mean-shift algorithm(B); the determined center (C).	30
Figure 3-8: Arcs at different ellipses merging one ellipse.....	30
Figure 3-9: Geometric schematic of parameter estimation.....	33
Figure 3-10: (A) Ellipse validation by the ratio of the length. (B) Ellipse validation via arc circumference ratio.....	35
Figure 3-11: The effectiveness values and execution times for the first four thresholds.....	39
Figure 3-12: The effectiveness values and execution times for the last four thresholds.....	39
Figure 3-13: The detection practical examples for Dataset Prasad (the 1 st three rows) Dataset #1(the last three rows).....	42
Figure 3-14: The detection practical examples for Datasets HX (the 1 st three rows) and DHX (the last three rows).	42
Figure 3-15: Real-time ellipse detection in a sequence of 214 images. A cluttered background with light and dark colours is used to verify the robustness of the presented detector.	44
Figure 3-16: Statistic results of detecting six ellipses in a video stream.	45
Figure 3-17: Detailed detection performance in all image frames for one of the objects. (F) shows the execution time for each image.	45
Figure 4-1: Robot grasping a soya milk carton occluded in a cluttered environment. The result of pose estimation is visualized by the green bounding box.....	48
Figure 4-2: The processing pipeline of our proposed system. (A) The Hough forest with contextual information; (B) the CNN module for refining the pose;	

the red arrows denote the input and the output for modules A and B; the blue dot indicates the output energy value.....	49
Figure 4-3: 3D models of the twelve textured and texture-less objects.....	51
Figure 4-4: Sampling the synthetic viewpoints in the upper hemisphere for object patch generation: the black hemisphere represents a virtual camera. The generated data from synthetic viewpoints are enclosed by the red circles.	51
Figure 4-5: The relation of numbers of the nodes at the current depth and the tree depth for 2,4,6,8 decision trees. Each broken line represents a tree.	56
Figure 4-6: The concept of depth to scale mapping.....	58
Figure 4-7: Detection for object class and pose by casting votes. A,B,C,D represent the voting spaces for object class and the pose including the position and the orientation, respectively. NMS means non-maximum suppression.....	58
Figure 4-8: Renders of textured or texture-less 3D mesh models(red) and the results of pose estimation visualized by the bounding boxes.	60
Figure 4-9: The output of each forest in a stack of forests with auto-context.	63
Figure 4-10: The CNN architecture.	65
Figure 4-11: Parameter determination and time comparisons. For (F), Wt (without reservoir sampling and distribution compression), RS (Reservoir sampling), DC (Distribution compression).....	68
Figure 4-12: The average results for self-comparison. CS (Contextual sensitivity), AC (Auto-context), L1 (L1-loss), L2 (L2-loss).	70
Figure 4-13: The comparison results for the methods with CNN and without CNN.	70
Figure 4-14: The examples of pose estimation.	72

Figure 4-15: The examples of pose estimation.	72
Figure 5-1: The setups of robotic grasping experiments.	76
Figure 5-2: Snapshots of dynamic detections at 40% of full speed of robotic motion for comparing the presented detector (-D) with Jia's detector (-J) and Fornaciari's detector (a,b,-F) at 0.2Hz sampling frequency.	77
Figure 5-3: Sequence of detected ellipses in a dynamic scenario with multiple moving cylindrical objects. The.....	79
Figure 5-4: Stacking cans in a dynamic scenario. Place the first can (1-A,B,C), stack the second can on the first can (2-A,B,C) and stack the third can on the second can (3-A,B,C).....	81
Figure 5-5: The robot grasping cans on the conveyor with 4 speeds in a dynamic scenario. The target is enclosed by a red frame. The speeds of A,B,C,D are 2.5cm,5.0cm, 7.5cm, 10.0cm,respectively. Several detection examples for different speeds in the middle column.	84
Figure 5-6: The steps of robot picking and placing cans. The target is enclosed by a red frame.	86
Figure 5-7: The steps of robot grasping an object (the first row) and robot grasping 12 different textured and texture-less objects in cluttered environments (the second and third rows). The target is enclosed by a yellow frame. The object pose is visualized by the green box representing 6D pose segmentation.	89
Figure 5-8: Robot grasping a soya milk carton (green 6D pose segmentation in the top row), a chip box (blue 6D pose segmentation in the top and middle rows), and a lemon box (red 6D pose segmentation) occluded in a cluttered environment. The target object is enclosed by the yellow frame in each row...	90
Figure 6-1: Objects enclosed by bounding boxes (A) and the dimensions of a set of objects approximated by the dimensions of bounding boxes (B).	95

Figure 6-2: A schematic of an under-actuated robotic gripper with TDMs. ...	97
Figure 6-3: (A) Passive tendon contacted a spring and (B) Active tendon driven by a motor.	99
Figure 6-4: Schematic diagram of the tendon-driven mechanism regarding an under-actuated finger. The purple circle on the right represents an actuator...	101
Figure 6-5: The schematic diagram of the geometric model for the enveloping grasp. The red dots represent the contact points.	103
Figure 6-6: The schematic diagram of geometric model for the fingertip grasp. The red dot represents the contact point.	103
Figure 6-7: Schematics of the enveloping angle (A) and forces exerting a surface along the tangent direction (B).	106
Figure 6-8: The tendon routes guided by the joint mandrel and pulleys. The blue dots represent the pulleys.	108
Figure 6-9: The grasping space of gripper with two 3-link fingers under different angle combinations.....	114
Figure 6-10: The initial state of the gripper to grasp an object (A). The gripper grasps the elliptical object (B) and the rectangle object (C).	116
Figure 6-11: The first joint: the objective values of different generations and the current optimized values.	116
Figure 6-12: The second joint: the objective values of different generations and the current optimized values.	117
Figure 6-13: The third joint: the objective values of different generations and the current optimized values.	117
Figure 6-14: The designed CAD model (A) and the prototype gripper (B) ..	118

Figure 6-15: The practical grasping cases. 119

List of Tables

Table 3-1: The Arc Extraction Algorithm	24
Table 3-2: Comparison results for four datasets. F, T, P and R represent the F - <i>measure</i> , <i>Times(ms)</i> , <i>Precision</i> and <i>Recall</i> , respectively. M, L, B, P, F, J, D represent the methods proposed by Mai et al., Bai et al., Prasad et al., Fornaciari et al., Jia et al., DED et al., respectively.....	41
Table 4-1: The detail algorithm of regression models.....	56
Table 4-2: Results of F-measure.....	72
Table 4-3: Results for Metric 1.....	72
Table 5-1: Comparison results for various speeds.....	79
Table 5-2: Experimental results of robotic grasps in dynamic scenarios	84
Table 5-3: Experimental results of robotic grasps	89
Table 6-1: Dimension constraints for robotic gripper.....	110
Table 6-2: The first joint: ranges of the parameters.....	112
Table 6-3: The second joint: ranges of the parameters	112
Table 6-4: The third joint: ranges of the parameters.....	112
Table 6-5: Geometric dimensions, average optimized results and final values	116
Table 6-6: The first joint: the optimized results.....	118
Table 6-7: The second joint: the optimized results	118
Table 6-8: The third joint: the optimized results	118

Chapter 1 Introduction

1.1 Motivation

Robots in household or warehouse scenarios need to have the ability to grasp for performing picking and placing tasks. Yet it is unstructured for household or warehouse environments since the information of these circumstances is uncertain, irregular and difficult to be described. This may lead robots to have difficulties in grasping objects in such environments.

Robotic manipulation employs object detection in images to create scene awareness and locate the objects' pose. Cylindrical objects are characterized by ellipses in the images depicting the circular top of cylinder at most angles of observations. Many objects commonly found in households (e.g. cans, cups, bottles, batteries, handles, candles) and industrial environments (e.g. plastic/metal caps, pipes, screw holes) are represented by cylindrical shape, whose fast detection and tracking could play an important role in manipulation and grasping strategies, particularly for dynamic scenarios. As different grasp types are needed for different objects, some researchers define grasp strategies based on the shape primitives, i.e., cylinders, boxes or spheres. For example, most of the cylindrical objects can be grasped with a lateral cylindrical wrap or top grasp using two or more fingers, depending on the type of the gripper, while other strategies such as prismatic grasp is preferred for a cubic shape object. As the cylindrical shaped objects appears widely in manipulation applications and is a key shape primitive, they have gained a considerable attention in research and development of both shape-based visual recognition methods and also shape-primitive based grasp strategies.

Thus, the tracking and robotic manipulation of cylindrical objects in dynamic scenarios can be enabled by fast ellipse detectors in the image streams in real-time. Although this approach hugely simplifies the problem of creating robotic scene awareness in the context of cylindrical objects, the problem of ellipse detection for such applications is challenging due to the following reasons. The manipulative action of robot implies that the scene acquired by robot's camera

changes dynamically and the object of interest may get occluded behind other objects or may appear overlapping with other objects. Detection of ellipses in real-time, in complex scenes, and in the presence of occlusion or overlaps is challenging. One of motivations in this work is to provide available solutions for these cases.

As robotic systems are getting employed in unstructured environments, they are required to manipulate textured and texture-less objects in highly cluttered and occluded scenes. Performing robotic manipulation is an ongoing and critical challenge, for which perception has been the main bottleneck. A typical scenario involves pick-and-place tasks, where a robot has to estimate the pose of an object in a cluttered environment, pick up the object, and move it to a designated position. A large number of efforts are invested in tackling the pose estimation problem. Although the associated challenges have been partly handled in previous works, occlusions among multiple objects can make pose estimation very difficult from a single viewpoint, supporting several ambiguous pose hypotheses. The inaccurately estimated poses greatly limit the ability and generality of robots to reliably make decisions and carry out manipulation actions, especially in complex environments with considerable clutter. Moreover, even with an appropriate image descriptor, few of the approaches can run at high speed because of large collection of relevance vectors.

From our literature review, many research groups mainly focus on the algorithm improvement of the perceptive system to make up for the shortcomings of the end-effector design that has less picking capabilities. However, the geometric grasping property of the gripper is a key factor for the robotic grasping system to perform well.

The design optimization of a robotic gripper is of utmost importance for achieving a stable grasp behavior. The robotic grippers are required to be all-purpose and capable such as to perform stable grasps and manipulations in unstructured environments. High cost and high complexity design cannot enable robotic grippers to be refined into products. To moderate these limitations, researchers have proposed under-actuated robotic grippers with

tendon-driven mechanisms (TDMs). Robotic grippers with two-finger and single actuator are prevalent in research and industrial applications because of their effective grasping capabilities, mechanical simplicity and low cost. However, robotic gripper design is a very complicated process involving modelling with many parameters. The manipulation performance of an under-actuated gripper significantly depends on the design rather than the control method.

Thus, in order to cope with these issues in robotic perception and grasp in unstructured environments, we embark on the study of tracking, grasping of static and dynamic cylindrical objects via the ellipse detection. We explore a high efficient learning approach to estimating 6D (Degree of Freedom) pose of the textured or texture-less objects for grasp purposes in a cluttered environment where the objects might be partially occluded. It is essential for presenting a novel versatile optimization design for two-finger grippers with TDMs so that the optimized gripper can realizing stable grasps in household and office environments.

1.2 Objective

In this thesis, I aim to develop a novel system of robotic perception and grasp. I propose the corresponding methods that can detect ellipses, track and grasp cylindrical objects under static and dynamic situations, estimate the poses of multiple objects and occluded objects in cluttered environments, as well as optimize design of underactuated robotic gripper with multiple joints. The robotic system should be applicable to daily grasping applications in households, warehouses and automatic production lines. The system should have the following characteristics:

- 1) In order to detect ellipses in images, track cylindrical objects, grasp static and dynamic cylindrical objects, the proposed algorithm of ellipse detection should be not only accurate but also fast.

- 2) The system should be able to estimate poses of multiple objects and occluded objects in cluttered environments so that the robot can realize grasping performances.
- 3) The under-actuated robotic gripper should be able to stably grasp objects with a large range of sizes. Therefore, the dimension parameters and transmission characteristics of the gripper need to be optimized.

1.3 Organization of Report

The remaining chapters are organized in the following manner:

Chapter 2 provides the literature review.

Chapter 3 introduces a real-time robust ellipse detection method for detecting cylindrical objects for robotic manipulation. The method takes the information of the edge curvature to split curves into arcs and classify them into different quadrants of a candidate ellipse, which further allows robust selection of arcs that may belong to ellipses. The proposed ellipse detector is verified by performance comparison with several state-of-the-art algorithms using four public image datasets.

Chapter 4 considers the issue of robotic grasps for given poses of textured and texture-less objects from an RGB-D video stream of a scene. Given a single RGB-D image, we use random forest integrated by the contextual information to deduce the object class probability and cast votes for the 6D pose in Hough space by joint classification and regression framework. Further, a convolutional neural network (CNN) is applied to choosing the best pose from the above stage. Extensive experiments on various public datasets indicate that our method is superior to several state-of-the-art works.

Chapter 5 describes the implementation of the algorithm on a robotic manipulator to investigate the feasibility of using ellipse detection to identify objects in real-time for grasping. Also, we experimentally demonstrate that our

method of estimating object 6D pose enables a robot to perform grasp capability on 12 textured and texture-less objects encountered in supermarket scenes stably.

Chapter 6 focuses on analyzing the optimal design of an under-actuated tendon-driven robotic gripper with two 3-phalange fingers and the proposed design optimization method to achieve a stable grasp performance. Practical experiments are performed as well to validate the proposed approach. The approach is universal for optimizing any conventional under-actuated tendon-driven gripper.

Chapter 7 highlights the contributions, and discusses about the future works.

Chapter 2 Literature Review

Over the past decades, research in robotic perception and grasp has flourished, and a great deal of effort has been spent on robotic perception strategy. Existing perception works for robotic grasp can be classified into two groups such as the touching perception and visual perception. Here we just discuss these important progresses that have been made toward applying visual perception techniques to the grasping problems, which are related to the techniques relevant to our work.

- Perception by 2D information- The perception based on 2D information is one of the more established fields of learning and there have been many successful applications to robotics. This discussion here focuses on the use of 2D image to robotic manipulators and then, in particular, the ellipse detections are reviewed since such perception strategy will be applied to tracking cylindrical objects, grasping cylindrical objects in static and dynamic scenarios.
- Perception by 3D information- Whereas the perception based on 2D information typically has the goal of succeeding at a robotic task, 3D perception has the explicit advantage in estimating 6D pose of object in the 3D space. A brief overview of the ideas and applications of pose estimation of an object is presented.
- Synthesis of robotic perception and grasp- By detecting the object and estimating the object's pose, the robot can manipulate objects in a practical life such as households as well as industries and the like. There has been significant prior work done in the use of visual perception for robotic manipulation tasks.
- Design optimization of under-actuated robotic gripper- The design optimization of a robotic gripper is of utmost importance for achieving a stable grasp behavior. This section discusses some of the major optimization methods used during the gripper design period.

2.1 Perception by 2D Information

In this thesis, we make use of ellipse detectors to realize the tracking and grasping of static and dynamic cylindrical objects. The methods of ellipse detection are reviewed briefly.

Five-dimensional Hough Transform (HT) is a classical method for detecting ellipses. Regardless of the robustness, the standard Hough transform [1] involves a large amount of computational cost. To improve the computational efficiency of HT, various HT-based approaches[2-6], such as Randomized Hough Transform (RHT) [5] and Probabilistic Hough Transform (PHT) [6], have been developed. However, these variants of HT cannot reach video rate computation speeds due to the process of voting among numerous candidates. Hybrid methods that combine Hough Transform and other geometric approaches have also been proposed to overcome the shortcomings of Hough Transform. For instance, Cakir et al. [7] combined feature-based models with HT to improve the detection accuracy. Chen et al. [8] integrated the advantages of HT and edge segment detection, described next, which allows its applications to industrial scenarios. However, it is still difficult for detection strategies based on Hough transform to reach a real-time detection speed.

Another class of ellipse detection approaches is edge segment detection techniques. Methods in this class exploit the connectivity between edge pixels to detect ellipses. The main steps of these methods are to extract arcs and then group them by exploiting geometric or algebraic properties of ellipses[9]. Specifically, edge arcs that may consist of an ellipse are found by variety of techniques such as the statistical regression method [10], curve segmentation by fitting a set of short line segments on edges [11], connectivity and curvature conditions [12, 13]. Subsequently, these arcs are grouped according to the convexity-concavity [9, 11, 14-16], arc curvature [15], and geometric constraints [17-20]. Lastly, ellipses are fitted on the grouped arc candidates using RANSAC [10, 21] or least squares fitting technique [14]. Prasad et al. [15] used arc convexity to determine search regions where suitable arc candidates for grouping may be located. Fornaciari et. al [14] used geometric

Figure 2-1: Several cases of ellipse detectors proposed by Prasad et al. (A), Jia et al. (B), Liu et al. (C), Mai et al. (D).

constraints as a selection strategy of arcs belonging to the same ellipse and estimate parameters by decomposing the parameter space. Mai et al. [12] adopted line segments to approximate potential elliptic-arcs according to connectivity and curvature conditions. Bai et al. [16] used the property of elliptical concavity to group arcs for identifying candidate ellipses. Liu et al. [20] proposed a hierarchical approach which is motivated by the fact that any segment of an ellipse can identify itself in ellipse reconstruction by geometric constraints. The edge segment detection methods are regarded as the most effective in detecting multiple ellipses in digital images and require less computational cost than HT-based methods. However, because of dependence on the preciseness of arcs detected, the detection accuracy drops when these methods are applied to images containing complicated occluded contours.

Many algebraic functions or geometric characteristics have been utilized for selection of arc segments to improve the performances of ellipse detectors. For instance, symmetry of arcs in image is applied for decomposing the parameter space and reducing the clustering parameters[22, 23]. Distance measures are used to perform registration of a set of arcs [22, 24, 25]. Mulleti et al. [26] relied on the sampling structure of finite-rate-of-innovation signals to estimate

the parameters of ellipses from partial data. Pătrăucean et al. [27] proposed a line segment and elliptical arc detector without any parameter tuning based on a contrario theory. Chang Liu et al. [28] fitted the elliptical projections of the cross sections of a surface of revolution with the given intrinsic matrix of a camera. Jia et al. [29] used a projective invariant operator to significantly prune the undesired candidates and pick out elliptical ones. Points and tangents together are regarded as the basic units of information to find an ellipse. The aforementioned methods can detect ellipses with less bias, but the performances of these methods are inadequate to detect occluded and overlapping ellipses when input images contain a large amount of noise. Several detection cases are illustrated in Figure 2-1.

2.2 Perception by 3D Information

There exists a vast amount of research activities regarding 3D object pose estimation, with the most representative works presented in literature generally subdivided into template-based and feature-based. We briefly introduce techniques that specifically provide some solutions of the pose estimation of objects in cluttered environments below.

The common approach for detecting object and predicting coarse pose is template matching, e.g.[30-33]. Due to template matching based on 3D descriptors, such methods can filter wrong 3D correspondences to deal with occlusion issue in a cluttered environment. A distance between the explored template and the template attached on an object is measured to realize the best match to estimate an accurate 6D pose. pose estimates using ICP(Iterative closest point). The authors integrate a scale-invariant patch template into a random forest by means of a new split function in a regression way, which provides excellent occlusion segmentation masks for obtaining object pose information [34]. Using template-based matching, LineMOD [35] performs robust object detection based on extracted patches from 3D object models. Rios-Cabrera et al. [36] stem from the template-based strategy and extended from learning the templates in a discriminative fashion to obtain the object's

Figure 2-2: The pose estimation methods from Brachmann et al. (A), Doumanoglou et al. (B), Kehl et al. (C), Choi et al. (D).

pose. The estimation procedure adopts the sliding window paradigm to search the template candidates that are then confirmed by matching feature points for estimating the pose [37]. Template-based methods demonstrate superior performance, especially for texture-less objects [35]. Template-based techniques can work accurately in practical applications, but also suffer in cases of occlusions and overlaps. Specifically, the performance of such methods degrade significantly when objects have poor informative shape since they match complete template with a target one by “global” features. Several methods that adopt patch-based [38] or pixel-based features [39] with a random forest classifier are proposed to deal with the above issues. Choi et al.[40] used the geometric features to match the templates of 3D object models for estimating 6D pose.

Dense features have been utilized for improving the performances of pose estimation[41]. In terms of dense features, it means each pixel in image generates some predictions corresponding to the desired outputs. Hough voting approach in a prediction space is employed to all pixels to explore the cell with

the most votes. Hough voting with small variance are used for predicting object pose, which achieves high detection accuracy [38, 42, 43]. RGB-D patches are applied to voting for the pose candidates of everyday objects rather than consider global information. The performance of such approaches is efficient when occlusion and clutter occur in the scene as the methods based on dense features just rely on partial context to obtain the object pose. Brachmann et al. [39] used a single RGB image to estimate object pose by means of an intermediate representation formed by two dense labellings such as the object coordinate and class. Brachmann et al. [44] model and exploit the uncertainties over 3D object coordinate in continuous distributions and over object labels in discrete distributions by the iteration strategy based on the auto-context [45] for estimating 6D pose. After predicting object class and coordinates depending on a classifier, Michel et al [46] formulated a novel Conditional Random Field for getting the pose candidates to generate global hypotheses for refining object pose. In fact, approaches that extract dense features have also one fundamental shortcoming. The chosen features via Random Forests can produce great influence on the number of votes to cast, which results in the required computational effort. Recently, Brachmann et al. [5] introduce a novel representation based on a combined coordinate and class labelling of 3D object, which, however, becomes invalid due to occlusions.

Currently, CNNs have been applied to learning RGB-D or RGB features [47-49]. Brachmann et al. [39] adopted a random forest to vote on extracted patches for detecting an object and estimating its pose based on learning RGB-D represents. A convolutional auto-encoder is employed to train random local patches depicting a scene to match against a database that contains patches extracted from 3D models [50]. Krull et al. [51] used the reinforcement learning to optimize the parameters of a recent work regarding the pose estimation, where the system of pose estimation is a stochastic policy, parametrized by a CNN [52]. Depending on the Euclidean distance, Wohlhart et al. [47] can assess the similarity between the descriptors. Specifically, a designed convolutional neural network is trained to compute these descriptors via the similarity and dissimilarity constraints in the descriptors. Kehl et al. [48]

extended the popular paradigm to cover the full 6D pose space and train on synthetic model data only. Balntas et al. [49] employed exact pose differences between the training samples to obtain the object’s pose by learning robust features. The main limitation of learning-based approaches is that they generally have to learn the background considered as the corresponding negative class, making the performance determined by the dataset classification.

In the brief review below, we discuss previous works related to the techniques relevant to our work, i.e., Hough forest, contextual information.

Random forests [53] have been used in several computer vision applications [42] because of their ability to deal with large training datasets and fast computation [54]. Recent works make use of random forests that map image features to vote in the generalized Hough space [54-56]. Hough voting approach is employed to all pixels to cast votes in Hough space and the cell with the most votes is used. Hough voting with small variance is used for predicting object pose, where small variances indicate high detection accuracy [38, 42, 43]. Doumanoglou et al. [55] proposed an unsupervised feature learned from depth-invariant patch samples by means of a Sparse Auto-encoder and offered an extensive evaluation of a variety of features that are used in voting in Hough space for estimating object pose. Latent-Class random forests in Hough space utilize clutter and occlusion information from inference stage to update the distribution of the latent object class and voting results jointly and iteratively [56], which efficiently improves the estimation accuracy. Differ from [42], we use the elements of the feature vector as an evaluation criterion for splitting the decision tree rather than determining the offset from the centroid of the bounding box to the center of the patch.

Context information is receiving attention gradually in the field of scene understanding [42, 44, 57-60]. Context sensitivity is a classical approach to boosting learning classifier depending on the image appearance and contextual information iteratively [60-62]. A feed-forward approach that includes context with appearance is proposed in [63] for object detection. Contextual

information has been used in the aspect of object segmentation [59], primarily for reasoning with the high level in random fields and to deal with contradicting segmentation results. Several algorithms [57, 64, 65], which are directly related to auto-context, integrate contextual beliefs that are considered as a weak learner into the boosting algorithm. As for the auto-context algorithm, the context information is integrated into rich image appearance via learning a variety of classifiers, introduced in [45]. This directly aims the posterior via iterative steps, leading to more efficient and effective algorithm. Auto-context has been widely applicable to object pose estimation [44, 51, 66, 67]. Several cases of related works are illustrated in Figure 2-2.

2.3 Synthesis of Robotic Perception and Grasp

A comprehensive literature review of robotic grasp is presented in [68, 69]. Early pose estimation for robotic grasp recovers object pose by a set of 2D keypoint correspondences [70, 71] with respect to textured objects, and the preferred edge features for less textured objects [72-74]. Since a RGB-D sensor is available at low cost, pose estimation based on 3D descriptor is more feasible than ever[75-77]. This work further combines local keypoint descriptor, shape, depth, and scene semantic as features [78-81]. Learning approaches have been proven effective in a wide range of robotic perception systems, making the perception to learn a mapping from a variety of features to various visual properties. The robots, so far, realize grasps by estimating a partial pose that includes less than 6 degrees of freedom [82-86]. Recent works have used ‘grasping point’ that is estimated from incomplete pose estimated from an image using supervised learning algorithms, allowing robots to grasp known objects with an unknown pose [87] and unknown objects for which the system has not trained, which was

Figure 2-3: The grasp systems of Jiang et al. (A), Levine et al. (B), Lenz et al. (C), Saxena et al. (D), Dmitry et al.(E) and Jeffrey et al. (F).

proposed by [88]. The representation suffers from a main limitation: the representation of their grasping point just indicates where to grasp, as well as the orientation is left to be calculated via approaches described next. The grasp synthesis problem is formulated as finding a successful grasp configuration, which is a planar position and an orientation visualized by rectangle representation [83, 89]. Jiang et al.[89] and Lenz et al. [85] used surface normals to obtain the approach vector of grasping objects after representing a grasp configuration as an oriented rectangle in 2D image space. Although the estimated pose is imperfect and inaccurate, it allows the robot to grasp an object in real time. However, this method is not suitable for manipulations with high precision positioning requirement. Levine et al.[90] and Dmitry et al.[91] utilized many robots for collecting the large-scale data to train the grasping models. Jeffrey et al. planned robust grasps with synthetic point clouds and analytic grasp metrics [92]. Some grasp strategies are shown in Figure 2-3.

2.4 Design Optimization of Under-actuated Robotic Gripper

This pioneering works for under-actuated tendon-driven robotic grippers and the corresponding mathematical models are briefly reviewed here. Hirose and Umetani [93] demonstrate the capabilities of an under-actuated tendon-driven

robotic hand. The under-actuated mechanism has been utilized in most of robotic hands, such as RTR2 Hand [94], SDK hand [95], and Velo gripper [96], Willow Garage hand [97], robotic gripper with soft surfaces and underactuated joints [98], GR2 Gripper [99], the Model S hand [100], the DEXMART hand[101], the hand of the DLR Hand Arm System[102], tendon-driven index finger[103], Soft-touch gripper [104]. Some models are shown in Figure 2-4. In addition, there exist many classical publications describing the kinematic and dynamic models regarding an under-actuated tendon-driven gripper. Tsai et al. [105] had established the mathematical model regarding the kinematic structure of tendon-driven robotic mechanisms by the graph theory. The identification and enumeration of the kinematic structure of tendon-driven robotic mechanisms are introduced using a pseudo-triangular structure matrix by [106]. Ou and Tsai [107] proposed a methodology of driving design equations regarding kinematic synthesis of tendon-driven manipulators based on isotropic transmission characteristics. A tendon-driven mechanism (TDM) with active and passive tendons can be grouped into several classes by kinematic analyses, which is presented by Ozawa et al. [103, 108]. However, the prior works do not present the relation between the active force and the contact forces according to the kinematic structure of TDMs and also do not present an approach to calculating the Jacobian of the structure of tendon-driven mechanism using the specific parameters.

The representative optimization methods are given for the under-actuated gripper design. The non-dominated sorting genetic algorithm version II (NSGA-II) is adopted to optimize the force extracted by the robot gripper on the surface of a grasped rigid object in [109]. Datta et al.[110] used multi-objective evolutionary algorithm to optimally calculate the dimensions of links and the joint angle of a robot gripper. Backus et al.[100] presented the optimization design approach of an underactuated robotic gripper by comparing the grasping performances of the cylindrical fingers and the single joint fingers. Ciocarlie et al. [111] built a function determining the size range of objects to optimize the links of the gripper. The dimension of robotic gripper is optimized via teaching-learning-based algorithm provided by [112]. The dimension

Figure 2-4: The Velo gripper (A); the Model S hand (B); the GR2 Gripper (C); the DEXMART hand (D); the hand of the DLR Hand Arm System (E); tendon-driven index finger (F).

optimization of the robotic gripper is presented by Dollar and Howe [113], depending on the grasping scenarios. Gosselin et al.[114] optimized the route of the active tendon routes by calculating the contact forces for a given configuration. The optimized method, which is based on the line constraint and the plane constraint, is proposed by [115]. Ciocarlie et al. [116] provided an approach to optimizing the tendon route such as to make the moment arms be well controlled.

2.5 Summary

This chapter reviews about some of the extensive prior work done regarding the topic of robotic perception and grasp. From the literature review, many impressive results have been obtained but there are still substantial open research questions. Too many prior structures can significantly constrain what it is possible for the robot to grasp by the perception. For ellipse detection, tracking of cylindrical objects, grasping of cylindrical objects in static and dynamic scenarios and pose estimation of multiple objects and occluded objects,

the optimal design of under-actuated robotic gripper, there are still many research problems that remain unresolved. For visual perception, the main challenges are as follows. First, the perception speed is not real time, which limits the use of visual perception in robotic manipulations. Second, the performances of detection and estimation accuracies are low. As for the design of an under-actuated tendon-driven gripper, the optimizations of more than twenty parameters and tendon routes are not still derived very well. Therefore, a systematic study on the robotic grasping system, the optimization design of under-actuated gripper and the corresponding methodology need to be conducted.

Chapter 3 Visual Perception based on Ellipse Detection

Cylindrical objects are characterized by ellipses in the images depicting the circular top of cylinder at most angles of observation, see Figure 3-1 for an example. Many objects commonly found in household (e.g. cans, cups, bottles, batteries, handles, candles) and industrial environments (e.g. plastic/metal caps, pipes, screw holes) are represented by cylindrical shape, whose fast detection and tracking could play an important role in manipulation and grasping strategies, particularly for dynamic scenarios. As different grasp types are needed for different objects, some researchers define grasp strategies based on the shape primitives, i.e., cylinders, boxes or spheres. For example, most of the cylindrical objects can be grasped with a lateral cylindrical wrap or top grasp using two or more fingers, depending on the type of the gripper, while other strategies such as prismatic grasp is preferred for a cubic shape object. As the cylindrical shaped objects are one of key shape primitives that are widely appeared in manipulation applications, they have gained a considerable attention in research and development of both shape-based visual recognition methods and also shape-primitive based grasp strategies.

Thus, the tracking and robotic manipulation of cylindrical objects in dynamic scenarios can be enabled by fast ellipse detectors in the image streams in real-time. Although this approach hugely simplifies the problem of creating robotic scene awareness in the context of cylindrical objects, the problem of ellipse detection for such application is challenging because of the following reasons. The manipulative action of robot implies that the scene acquired by robot's camera changes dynamically and the object of interest may get occluded behind other objects or may appear overlapping with other objects. Detection of ellipses in real-time, in complex scenes, and in the presence of occlusion or overlaps is challenging.

Figure 3-1: Tracking ellipses in a dynamic scenario.

Ellipse detection in images has been deemed useful in other applications as well, including industrial inspection [17], medical diagnosis [23], recognition of traffic signs [117], security [118], autonomous navigation[119], tracking targets [120, 121], and even manipulating cylindrical objects[75, 122, 123]. However, the current methods are marred with slow execution due to heavy computational loads and low ellipse detection accuracy. In this work, we propose a novel ellipse detector that performs better than the state-of-art methods in terms of execution speed and detection accuracy.

In this chapter, a novel algorithm to perform fast and robust ellipse detection with good detection performance is presented. The balance of precision and recall is achieved by extremely well-designed clustering approach that leverages heavily on two main geometric properties, namely edge pixel gradient and arc convexity-concavity. These geometric properties are calculated once and re-used across several steps, helping in improving the computation time. Clustering, parameter detection, and filtering are also performed in computationally inexpensive manner by employing simple computations and decision mechanisms throughout. Further, we incorporate filtering through out the algorithm at various levels, such as exclusion of almost linear arcs, very small arcs, stringent and refined definition of arc clusters based on relative convexity of arcs, and finally ellipse filtering based on saliency and redundancy.

Employing filters at different levels reduces the computational burden on subsequent steps and makes the ellipse detection more robust. As a consequence, the proposed algorithm can be used in real-time real-scenario video imagery for robotic manipulations.

The remaining content is organized below. In Section 3.1, the arc segmentation and classification are discussed. The approaches to detecting and validating an ellipse is described in detail in Section 3.2. Section 3.3 describes threshold confirmations and comparisons of the proposed method with six existing methods for four datasets. Lastly, the work is summarized in Section 3.4.

3.1 Arc Segmentation and Classification

3.1.1 Pre-processing Stage

The pre-processing stage helps to speed up the subsequent processes and improve the computational efficiency. First, Canny edge detector [124] with auto-thresholding is applied to obtaining the edge image in Figure 3-2. In an edge image, we analyze an edge point by the coordinates (x_i, y_i) and the phase of the gradient η_i at a pixel p_i . Every edge point p_i is classified into two main directions based on the gradient η_i . Due to the discrete characteristics of the digital image, it is difficult to obtain the accurate values of η_i [125]. Therefore, we use only the sign of the gradient as representative of the direction of the gradient rather than the Sobel derivatives [14]. The gradient sign function $X(p_i)$ at the pixel p_i is defined as follows,

$$X(p_i) = \begin{cases} +, \tan(\eta_i) > 0 \\ -, \tan(\eta_i) < 0. \end{cases} \quad (3-1)$$

When the Sobel derivatives dx and dy are zero along the vertical and horizontal gradient direction, we discard these points. To efficiently obtain valid arc combinations for later processing, we class the extracted arcs into four sets and correspondingly distribute them in four quadrants of one ellipse such as I, II, III and IV that represent the number of the first, second, third and fourth

Figure 3-2: The image by a pre-processing. The initial image (A); the edge images along two edge directions by gradient signs (B, C); the edge images after the process of 8-connectivity (D, E); the arcs enclosed by oriented bounding boxes (F, G); the image after pre-processing (H).

quadrants, respectively. In the pre-processing step, we have classed edge point p_i ($i = 1, 2, 3, \dots, n$) consisting of arcs into two groups based on the sign function $X(p_i)$. Consequently, the edge points with the positive gradient directions rest on the first and third quadrants ($p_i \in \{I \cup III\}$) while the other arcs belong to the second and fourth quadrants ($p_i \in \{II \cup IV\}$), as illustrated in Figure 3-2. Subsequently, the detector of the 8-connectivity of two consecutive edge points is adopted to remove edge curves with less than T_n since short curves may be noises or of no effect on detecting ellipses due to the lack of curvature. We use an oriented bounding box with the minimum area, which is denoted as OBB_{min} , to enclose a curve [126]. If the minimum side length of OBB_{min} is less than a threshold T_o , we can discard this curve since it may be a line segment. The remaining curves are deemed to be arcs that may represent ellipses and used in the subsequent steps for ellipse detection.

3.1.2 Arc Extraction

It is well known that the curvature change over an elliptic arc is continuous and smooth. We obtain such arcs by splitting curves at sudden changes in curvature in terms of the amount of change or the direction of change. We can define these two cases of sudden changes as follows. A turning corner is the point where there is a large change of curvature and an inflexion point is the point where the direction of curvature changes. We fit an edge curve approximately by a series of line segments so that the curve can be denoted as $\{l_1, l_2, \dots, l_N\}$ by the proposed method in [9, 127, 128], as shown in Figure 3-3(A). The approximation of the line segments fitting a curve reduces the computational cost since the subsequent calculation is only performed on the dominant points of the curve (i.e. the endpoint of the line segments of the curve) instead of all the points. The vector angle between consecutive line segments are defined as θ_i ($i = 1, 2, \dots, n$) whose direction is from l_i to l_{i+1} and the range lies in between $-\pi$ and π . We set a threshold (T_θ) to evaluate the amount of the change between the i^{th} angle θ_i and the $(i + 1)^{\text{th}}$ angle θ_{i+1} . Thus, turning corners are determined via

$$|\theta_{i+1} - \theta_i| > T_\theta.$$

Figure 3-3: A curve fitted by a series of line segments. Line segments with arrows represents the directions of curvatures of arcs and the red rectangle frames denote bounding boxes enclosing arcs extracted.

T_θ is set experimentally to be sufficiently high to conclude a large change or direction change in a curvature. If the difference absolute value of two consecutive angles (θ_i, θ_{i+1}) is larger than the threshold T_θ , the point p_{i+1} is regarded as a turning corner for the same directions. For example, an arc is extracted by splitting an arc at a change of direction of curvature p_4 and a large change of curvature p_7 (see Figure 3-3(B)). In this case, $|\theta_{i+1} - \theta_i|$ will also be larger than T_θ and such points p_i are determined to be the inflexion points. After extracting arcs, we use the method mentioned above in the pre-processing stage to remove curves that do not satisfy two geometric constraints. The algorithm of arc extraction, via splitting a curve at turning corners and inflexion points, is given in Table 3-1.

3.1.3 Arc Classification

After arc extraction, a series of arcs are generated. Arcs that consist of an ellipse must satisfy the corresponding convexity-concavity for each other. Each arc has a unique convex or concave property. Thus, all arcs can be classified into two classes based on the convexity-concavity. For the classification of arcs,

Table 3-1: The Arc Extraction Algorithm

Arc Extraction by splitting a curve at sudden changes

Input: T_θ , curves

Output: arcs that may consist of ellipses in a container C

For $i = 0$ to the size of curves **do**

Fit curve i by line segments;

Calculate the angles θ formed by consecutive line segments

For $j = 0$ to the size of line segments **do**

If ($\text{abs}(\theta_{j+1} - \theta_j) > T_\theta$)

For $m = 0$ to the size of the i^{th} curve **do**

If ($\text{sign}(\theta_j) \cdot \text{sign}(\theta_{j+1}) < 0$)

If (line segments $[j].x/y = \text{curve}_i[m].x/y$)

Reserve these points in the container B ;

Else

If (line segments $[j + 1].x/y = \text{curve}_i[m].x/y$)

Reserve these points in the container B ;

For $k = 0$ to the size of B **do**

For $n = B[k]$ to $B[k + 1]$ **do**

Reserve curve $[i][n]$ in the container C .

Figure 3-4: Arc classification based on the convexity-concavity.

[14, 129] use the difference between the upper and lower areas to determine the convexity-concavity of arcs. Actually, it is not accuracy when the case of Figure 3-4(C) occurs when their methods do not include the split strategy at turning corners or inflexion points. Here we propose an approach to determining the convexity-concavity of an arc. Specifically, if the midpoint position of an arc is higher than the midpoint position of a line segment formed by the endpoints of an arc, the arc is convex. Otherwise it is assigned to a concave arc. Differ from [14, 129], the proposed method can avoid the case of Figure 3-4(C) because of splitting curves at the turning points and inflexion points. As shown in Figure 3-4, we use A_u and A_l to represent the upper area and lower area, respectively; there are three cases, $A_u > A_l$ (Figure 3-4(A)); $A_u < A_l$ (Figure 3-4(B)); $A_u = A_l$ (Figure 3-4(C)); P and Q denote the midpoints of the arc and the line segment formed by the two endpoints of arc, respectively. This avoids the accumulation process of pixels for obtaining the areas to speed up the efficiency of the classification of arcs. The convexity-concavity of the arc e is determined by the function $\Theta(e)$ as follows

$$\Theta(e) = \begin{cases} +, & \text{convex;} \\ -, & \text{concave.} \end{cases} \quad (3-2)$$

When these two midpoint positions are the same as each other, such arc cannot be considered as a convex or concave arc, thus is rejected.

Based on the function X and Θ , each arc e can be classified to the specific quadrant. For example, the arc that belongs to $I \cup III$ can be further classed to the first quadrant (I) if $\Theta(e)$ is positive otherwise ($\Theta(e) < 0$) falls in the second

quadrant (II). In other words, when satisfying $X(p_i) > 0$ and $\Theta(e) > 0$, the arc e belongs to the quadrant set I. Under the conditions of $X(p_i) > 0$ and $\Theta(e) < 0$, the arc e is classed into the quadrant III, as shown in Figure 3-5. Where e represents an arc $\{Connected(p_{i-1}, p_i)\}$ and $Connected(p_{i-1}, p_i)$ verified by the 8-connectivity of two consecutive edge points.

3.1.4 Arc Grouping Based on Geometric Constraints

A set of three arcs $\tau_{abc} = (e_a, e_b, e_c)$ are chosen as a triplet depending on the below grouping constraints, which indicates that the arcs likely belong to the same ellipse. Accordingly, four combinations of a triplet of arcs that do not belong to the same quadrant are made, as follows: $(e_a, e_b, e_c) \in \{(I, II, III), (I, II, IV), (II, III, IV), (I, III, IV)\}$ where a, b, c are the indexes of different arcs, as shown in Figure 3-5.

We choose a pair of arcs $p_{ab} = (e_a, e_b)$ from the two subsequent quadrants such as $(e_a, e_b) \in \{(I, II), (II, III), (III, IV), (IV, I)\}$ according to the counter-clockwise order. Thus, a triplet is made up of two pairs of arcs sharing an arc, $\tau_{abc} = \{(p_{ab}, p_{cd}) | e_b \equiv e_d\}$. We use the grouping approach based on the relative positions among arcs to sort arcs sets. For instance, as depicted in Figure 3-5, e_i^l and e_i^r denote the left and right endpoints of an arc, respectively. For instance, along the horizontal axis, the coordinates of e_1^l and e_4^l must be larger than those of e_2^r and e_3^r . Similarly, along the vertical axis, the coordinates of e_3^l and e_4^r must be smaller than that of e_2^l and e_1^r (see Figure 3-5). A tolerance between two sides of inequalities is regarded as one because the pixels on the coordinate axes have been discarded. The locations of centres of such arcs are also considered as additional constraints for selecting potential arcs. If the centres of such arcs in a triplet rest on the same area within a range, it is highly likely that they fit the same ellipse. Sometimes, arcs belonging to different ellipses may merge together to fit one ellipse. The constraint of the distances between midpoints of arcs and the centres of arcs are applied to avoiding false detections.

Figure 3-5: Arc classification.

3.2 Ellipse Detection

3.2.1 Confirmation of Elliptic Center

The geometric characteristics used for retrieving the centres of the ellipses are presented here. The geometric properties of points and tangents in ellipses is used for finding elliptic centres[2, 15]. The elliptic centre is the intersection of the lines connecting the midpoints of chords and the intersection formed by the tangent lines through the endpoints of arcs. Although the method can detect the centre of a small ellipse, the slopes of tangent lines could contain many errors due to the image noise and the digitization of the image[129].

To avoid the cases mentioned above, we propose a new method of determining the elliptic centre. The chosen arc pair whose arcs must satisfy two geometric constraints: (i) convexity-concavity and (ii) relative positions is applied to calculating elliptic centre in the context of the assumption that two arcs rest on the same ellipse due to the presence of noise and short arcs less than a certain length. The methods proposed in [2, 15] are ineffective. as shown in Figure 3-6(A). In Figure 3-6, P_i denotes endpoints of arcs e_a, e_b, e_c ; P_{ij} represents the intersection of the tangent lines through P_i and P_j . M_{ij} is the midpoint of the line segment P_iP_j ; l_{ij} indicates the line formed by P_{ij} and M_{ij} ; the shifted intersection points (yellow dots) are formed by the shifted lines (yellow lines)

joining the midpoints of the chords and the intersections of the tangent lines when the slope of the line through the midpoint P_{34} of the arc a_b is not correct due to image noise, $i, j=1,2,3,4,5,6$. In our method, the centre is estimated based on the maximum distances among endpoints of arcs for a pair of arcs. Specifically, endpoints of a pair of arcs can form $\binom{4}{2}$ line segments. The case shown in Figure 3-6(B) is considered as an example, we extract endpoints (P_1, P_3, P_4) forming the two longest line segments (P_1P_3, P_1P_4) to obtain the intersections (P_{13}, P_{14}) of tangent lines and midpoints (M_{13}, M_{14}) through endpoints (P_1, P_3, P_4) and then, the intersection called by the elliptic centre is generated by two lines (l_{13}, l_{14}) passing intersections (P_{13}, P_{14}) and midpoints (M_{13}, M_{14}) , as illustrated in Figure 3-6(B).

If the coordinate and gradient of endpoints of an arc are expressed as $\{x_i, y_i, \theta_i | i = 1, 2, 3, 4\}$, the coordinates of midpoints (M_{13}, M_{14}) of the line segments P_1P_3 and P_1P_4 are derived by Eq.(3-3). Similarly, the coordinates of intersections (P_{13}, P_{14}) of the two tangent lines through endpoints (P_1, P_3, P_4) are expressed in Eq.(3-4). We calculate the slope q_1 of the line segment P_1P_3 by Eq.(3-5) and the slope q_2 of the line l_{13} passing M_{13} and P_{13} is obtained through Eq. (3-6).

$$\begin{aligned} x_{M_{13}} &= \frac{x_1+x_3}{2}, y_{M_{13}} = \frac{y_1+y_3}{2}, \\ x_{M_{14}} &= \frac{x_1+x_4}{2}, y_{M_{14}} = \frac{y_1+y_4}{2}; \end{aligned} \quad (3-3)$$

$$\begin{aligned} x_{P_{13}} &= \frac{y_1-\theta_1x_1-y_3+\theta_3x_3}{\theta_3-\theta_1}, y_{P_{13}} = \frac{\theta_1y_3-\theta_3y_1+\theta_3\theta_1(x_1-x_3)}{\theta_3-\theta_1}, \\ x_{P_{14}} &= \frac{y_1-\theta_1x_1-y_4+\theta_4x_4}{\theta_4-\theta_1}, y_{P_{14}} = \frac{\theta_1y_4-\theta_4y_1+\theta_4\theta_1(x_1-x_4)}{\theta_4-\theta_1}; \end{aligned} \quad (3-4)$$

Figure 3-6: Illustration of the calculation method of the elliptic center. The methods of estimating the elliptic center proposed by others (A) and us (B, C). The red line segments are the two longest ones in all line segments(B).

$$q_1 = \frac{y_1 - y_3}{x_1 - x_3},$$

$$q_3 = \frac{y_1 - y_4}{x_1 - x_4}; \quad (3-5)$$

Figure 3-7: Procedure of a center estimation by mean-shift clustering algorithm. The intersections generated by two pairs of arcs (A); the center clusters enclosed by the dash circles with the mean-shift algorithm(B); the determined center (C).

Figure 3-8: Arcs at different ellipses merging one ellipse.

$$q_2 = \frac{y_{P_{13}} - y_{M_{13}}}{x_{P_{13}} - x_{M_{13}}} = \frac{(\theta_3 + \theta_1)(y_3 - y_1) + 2\theta_3\theta_1(x_1 - x_3)}{2(y_3 - y_1) - (\theta_3 + \theta_1)(x_1 - x_3)},$$

$$q_4 = \frac{y_{P_{14}} - y_{M_{14}}}{x_{P_{14}} - x_{M_{14}}} = \frac{(\theta_4 + \theta_1)(y_4 - y_1) + 2\theta_4\theta_1(x_1 - x_4)}{2(y_4 - y_1) - (\theta_4 + \theta_1)(x_1 - x_4)}. \quad (3-6)$$

Thus, the coordinates of the centre C^{ab} can be derived according to the proposed method as follows,

$$x_{C^{ab}} = \frac{y_{M_{14}} - q_4 x_{M_{14}} - y_{M_{13}} + q_2 x_{M_{13}}}{q_2 - q_4},$$

$$y_{C^{ab}} = \frac{q_2 y_{M_{14}} - q_4 y_{M_{13}} + q_2 q_4 (x_{M_{14}} - x_{M_{13}})}{q_2 - q_4}. \quad (3-7)$$

Using the same method, we can obtain the centre of a pair of arcs $p_{cd} = (e_c, e_d) | e_d \equiv e_b$. We consider that a set of arcs $\tau_{abc} = \{(p_{ab}, p_{cd}) | e_b \equiv e_d\}$

can consist of the same ellipse if and only if the distance of the centres of two pairs (p_{ab}, p_{cd}) lies within a given threshold T_c .

Theoretically, six intersections formed by a set of arcs τ_{abc} should be coincident with each other. However, due to the presence of noise and inaccurate edge angles, the geometric centres will shift into neighbourhoods of initial points such that these centres show a distribution in the parameter space of ellipse, as illustrated in Figure 3-6(C). Unlike conventional detection methods of ellipse centres by obtaining the median of the coordinates of a set of intersections in [14, 29], we also generate the centre whose coordinates are obtained by iterative mean-shift clustering algorithm [130] in Figure 3-7. In Figure 3-7(C), we can reach the converged centre that rests on an area with an acceptable range.

3.2.2 Distance Constraint for Ellipse Fitting

After determining elliptic centre, we have to choose the candidates of elliptic arc. In general, the distances between points on arcs and the elliptic centre should fall in a suitable range. For simplicity, we just use midpoints of arcs to replace other point on an arc for getting the distance between a point on an arc and the elliptic centre. We can eliminate these false positives via this geometric constraint. We define the following function $\Gamma(e_a, e_b, e_c)$ to describe the round-flat characteristics for discarding the false positive detections:

$$\Gamma(e_a, e_b, e_c) = \frac{\max(d_a, d_b, d_c)}{\min(d_a, d_b, d_c)} > T_f$$

where $\max(d_a, d_b, d_c)$ and $\min(d_a, d_b, d_c)$ are the estimated minimum and maximum values among three distances d_a, d_b, d_c between midpoints P_a, P_b, P_c of three arcs e_a, e_b, e_c and the centre C estimated in above subsection, respectively, as seen in Figure 3-8. If the centre is determined, we will further check the geometric constraint $\Gamma(e_a, e_b, e_c) > T_f$ before a triple of arcs is taken as members of candidates of elliptic arc. The threshold T_f is set to 5 in this

work. The approach cannot ensure that all the ellipses can be detected successfully as $\Gamma(e_a, e_b, e_c)$ of a triplet of arcs may be beyond the corresponding threshold T_f .

3.2.3 Parameter Estimations

To obtain the remaining parameters, we decompose the parameter space of an ellipse for the ratio N of the ellipse minor semi-axes length B to major semi-axes length A and other defined parameter K ($K = \tan \rho$, ρ is the orientation of an ellipse), as described in [2, 131]. The equation that expresses N in terms of K is provided as

$$N^2 = -\frac{(q_1-K)(q_2-K)}{(1+q_1K)(1+q_2K)}, \quad (3-8)$$

with $\alpha = q_1q_2 - q_3q_4$ and $\beta = q_2q_4(q_3 - q_1) + q_1q_3(q_4 - q_2) + (q_1 + q_2 - q_3 - q_4)$. Thus,

$$K = \pm \sqrt{1 - \frac{\beta}{\alpha}}. \quad (3-9)$$

For a pair of points found, the $N - K$ accumulator is continuously updated according to Eq. (3-8), we can get the highest peaks of the values of N and ρ are in two 1D accumulators. Here the ellipse polar equations and coordinate transformation relations are applied to transforming the coordinate (x_i, y_i) of a point on an arc in the world coordinate system to the coordinate (x_o, y_o) in the ellipse coordinate system. The coordinate transformation can be described by the following equations,

$$\begin{bmatrix} x_o \\ y_o \end{bmatrix} = \begin{bmatrix} \cos \rho & \sin \rho \\ -\sin \rho & \cos \rho \end{bmatrix} \begin{bmatrix} x_i - x_c \\ y_i - y_c \end{bmatrix}. \quad (3-10)$$

According to Eq.(3-8,3-9,3-10),we obtain the following equations,

the angular circumference ratio to discard the false detections. The calculations of a large number of angles of arcs lead to additional computational burdens. Obviously, outliers have effect on the detection of the endpoints of an arc, which results in the inaccuracy detections. Differing from [23], we adopt the ratio of half of the circumference of the bounding box enclosing an arc and the sum of the semi-axes lengths to measure the integrity of ellipse to improve the detection accuracy. This is significantly faster than other more complicated approaches of measuring the integrity of the ellipse. We give the specific verifications in the next section. Moreover, the ratio of the circumference is used to imply the closeness of arcs fitting an ellipse.

1) Validation by the ratio of the length

The ratio of the sum of the long and width lengths of bounding box enclosing an arc to the sum of the major and minor semi-axes lengths (A, B respectively) can be regarded as a measurement of circumference. Assuming that an ellipse is fit by the triplet $\tau = (a_1, a_2, a_3)$ enclosed by bounding boxes with the size of $m_i \times n_i (i = 1, 2, 3)$, we define a function $K(\tau)$ as follows,

$$K(\tau) = \frac{\sum_{i=1}^3 (m_i + n_i)}{3(A+B)}. \quad (3-13)$$

A higher value of $K(\tau)$ implies a larger probability of a candidate ellipse being a real one, as illustrated in Figure 3-10(A). If $K(\tau)$ exceeds a threshold (T_l), the ellipse candidate can be verified further by the below constraint, otherwise it is discarded. We define a score of $K(\tau)$ as $s(K(\tau))$, to represent the probability of a set of arcs consisting of a real ellipse.

2) Validation via the ratio of the circumference

Supposing that the coordinate of point on an arc is (x_i, y_i) , we put (x_i, y_i) into the following equations:

Figure 3-10: (A) Ellipse validation by the ratio of the length. (B) Ellipse validation via arc circumference ratio.

$$X = \frac{[(x_i - x_c) \cos \rho + (y_i - y_c) \sin \rho]^2}{A^2},$$

$$Y = \frac{[(y_i - y_c) \cos \rho - (x_i - x_c) \sin \rho]^2}{B^2} \quad (3-14)$$

where x_c and y_c represent the centre coordinates and ρ denotes the orientation of a detected ellipse. In Figure 3-10(B), the differential distance d between a detected elliptic arc and an edge of a real ellipse arc are given as follows:

$$d = |X + Y - 1|. \quad (3-15)$$

If d is less than a threshold (T_d), it means that the pixel (x_i, y_i) is close to the edge of the detected ellipse. The following function $\psi(\tau_{abc})$ is defined as the ratio of the number of a set of pixels $\mathcal{B}$ satisfying Eq.(3-15) to the number of pixels on the three arcs in the triplet τ_{abc} used for detecting an ellipse, which implies the closeness of arcs fitting an ellipse,

$$\psi(\tau_{abc}) = \frac{N_{\mathcal{B}}}{N_a + N_b + N_c} \quad (3-16)$$

where N_a , N_b and N_c are the numbers of pixels on the three arcs, respectively and $N_{\mathcal{B}}$ is the number of pixels in $\mathcal{B}$. If $\psi(\tau_{abc})$ is more than a threshold (T_{ψ}), the detected ellipse is considered as a valid one, otherwise this ellipse is given

up. The corresponding score of $\psi(\tau_{abc})$ is defined as $s(\psi(\tau_{abc}))$ for clustering.

3) Clustering by similarities among detected ellipses

Multiple candidate ellipses may be fit to form the same “real” ellipse as multiple arc sets may belong to the same “real” ellipse. In order to ensure a “real ellipse” is just made up of one arc set, such candidate ellipses are clustered depending on the order of the scores [15]. All valid ellipses with the same centre are ranked according to the total decreasing score $(s(K(\tau)) + s(\psi(\tau_{abc})))$. Then we assess the similarity of two ellipses by comparing the differences of ellipse parameters. Specifically, a feature vector is defined as $V(x_c, y_c, A, B, \rho)$ where (x_c, y_c) represents the coordinates of an ellipse center, A and B denote the lengths of the semi-major and semi-minor axes, respectively, ρ is the orientation of ellipse. The Euclidean distance between two feature vectors is considered as the distinctiveness measure:

$$D(V, W) = \sqrt{\sum_{i=1}^5 (E_{V,i} - E_{W,i})^2}, \quad (3-17)$$

where $E_{V,i}$ and $E_{W,i}$ denotes the i^{th} parameters in the vectors V and W . If $D(V, W)$ is less than a threshold T_s , the ellipses V and W are concluded to belong to the same ellipse cluster. Otherwise, the corresponding ellipse is regarded to be part of the new reference one. Thus, when an ellipse cannot be assigned to any cluster, it becomes a reference of a new cluster.

3.3 Threshold Confirmation and Detector Comparisons

We confirm all the thresholds by tuning these parameters, which allows our detector to realize a very efficient performance concerning speed, successful detection of multiple ellipses and to compare the presented detector with six state-of-the-art methods proposed by Jia et al.[29], Fornaciari et al.[14], Prasad et al.[15], Bai et al.[16], Liu et al.[20], Mai et al.[12]. To verify the advantages

of the proposed method, we use the same evaluation metrics in four public datasets with different characteristics, including Dataset Prasad [15], dataset#1[14], Dataset HX with a variety of complex backgrounds and Dataset DHX. As for Dataset HX, all the images with ellipses from real scenarios like households, industries, roads, are randomly chosen from 16 categories in the ImageNet repository [132] in Dataset HX. We also construct another dataset named Dataset DHX that contains images with cylindrical objects in household or office environments for testing the performance of a robot manipulating a cylindrical object by our real-time ellipse detector. These images are collected randomly by 20 volunteers. For taking each image, volunteers placed a camera approximately 50cm from the front edge of a platform and 50cm above the platform surface, tilted down by approximately 45° . Source codes of these methods in C++ or MATLAB are available online. All the experiments are performed on a PC with 8GB RAM and an Intel Core i7 processor.

3.3.1 Evaluation Metrics

Performance metrics, such as F -measure and the execution time t proposed in [15] and applied in [14], are used for quantitative comparison. For a detected ellipse ε_d and the ground-truth ellipse ε_g , the overlap ratio Φ is the Jaccard index of similarity between the ellipses ε_d and ε_g [15]. If the overlap ratio Φ satisfies $\Phi > \Phi_0$ with $\Phi_0 = 0.8$, the detected ellipse is considered a correct detection, otherwise, it is counted as a miss. As a result, the *Precision* and *Recall* values are computed as follows,

$$Precision = \frac{\Omega}{N}, Recall = \frac{\Omega}{M}, \quad (3-18)$$

where Ω denotes the number of correctly detected ellipses, N is the number of detected ellipses, and M is the number of ground-truth ellipses. Based on Eq.(3-18), F -measure is obtained as follows,

$$F\text{-measure} = \frac{2 \times Precision \times Recall}{Precision + Recall}. \quad (3-19)$$

3.3.2 Confirmation of Thresholds

Performance of the ellipse detection method mostly depends on the choice of threshold parameters, including $T_n, T_o, T_\theta, T_c, T_l, T_d, T_\psi, T_s$. However, we cannot ensure suitability of a chosen threshold in all images for achieving the best performance of ellipse detection. We can just choose a threshold that makes the best detection performance regarding the effectiveness *F-measure* and the execution time (t) by tuning a threshold parameter in the context of fixing other threshold parameters, as illustrated in Figure 3-11 and Figure 3-12.

When the threshold T_n is more than 15, the effectiveness *F-measure* drastically reduces while the execution time maintains a stable state roughly in Figure 3-11(A). T_n is therefore set as 15. The influences of the threshold T_o on the effectiveness and execution time are explored in Figure 3-11(B). It is quite unambiguous that the detection effectiveness does not show significant changes and the costing time becomes bigger and bigger when the threshold T_o is larger than 20. Thus, we choose 20 as the value of T_o . As observed in Figure 3-11(C), the best parameters are given with values of T_θ between 30 and 40 for *F-measure*. The execution time t decreases as values of T_θ increase. As a result, we set $T_\theta = 35^\circ$. As clearly illustrated in Figure 3-11(D), the best thresholds are 25 and 30 for the detection effectiveness, but the execution time of a 30 threshold is more time demanding than that of a 25 threshold. We choose $T_c = 25$ as a good trade-off to gain a good effectiveness and keep an execution time an acceptable value.

Figure 3-11: The effectiveness values and execution times for the first four thresholds.

Figure 3-12: The effectiveness values and execution times for the last four thresholds.

For a ratio threshold T_l larger than 0.6 a significant decline occurs in F-measure and the computational time also decreases due to the time decrease of the clustering step. We set $T_l = 0.5$ because this value can guarantee the best effectiveness and an acceptable execution time comparing with a value of $T_l = 0.4$, as depicted in Figure 3-12(A).

Figure 3-12(B) indicates that the detection effectiveness is basically stable, while the time used for detecting ellipses slightly increases as the threshold T_d improves. In this case, the best trade-off between the detection effectiveness

and cost time is confirmed with $T_d = 1$. The detection effectiveness dramatically decreases when the threshold T_ψ is greater than 0.5 because more arc sets are considered as invalid ones, while the fluctuation of the execution time is small in general, as shown in Figure 3-12(C). We adopt $T_\psi = 0.5$. In terms of T_s , it more than 25 has negative impact on the detection effectiveness, as illustrated in Figure 3-12(D).

Finally, we adopt $T_s = 25$. Thus, the thresholds are as follows, $T_n = 15$, $T_o = 20$, $T_\theta = 35^\circ$, $T_c = 25$, $T_l = 0.5$, $T_d = 1$, $T_\psi = 0.5$ and $T_s = 25$.

3.3.3 Performance Comparisons

The results of average *F-measure* values and average execution time are presented for four datasets in Table 3-2. The methods of Bai et al.[16], Liu et al.[20] and Mai et al.[12] present relatively lower performances, especially losing ellipses in images with complex backgrounds. Liu's detector has good execution speed but very poor effectiveness. Bai's method has better effectiveness for detecting large ellipses than Mai's method but still remains low. The method of Prasad et al.[15] outperforms all other methods in his own dataset (Dataset Prasad) while the proposed method is in the second place and close to Prasad's method in terms of the effectiveness *F-measure*. Prasad's detector is very time demanding though it shows relatively good effectiveness for detecting small and distorted ellipses. The results of execution time show that Jia's method [29] is the fastest one and its effectiveness is better than Fornaciari's method [14]. Fornaciari's and Jia's detectors show the excellent execution speeds but suffer with the amount of small, occluded ellipses given in images in detection effectiveness. In other three datasets, the proposed method is superior to other methods for the effectiveness, especially in detecting small ellipses and occluded and over-lapping ellipses. In terms of average execution speeds, the execution time of our approach is less than 1% of the execution time of Prasad's method and falls behind methods of Fornaciari and Jia. However, the detection speed of the proposed is still in an acceptable range for real-time applications. The practical cases for static nature images are provided in Figure 3-13 and Figure 3-14. Note that our proposed ellipse detector is named as DED.

Table 3-2: Comparison results for four datasets. F, T, P and R represent the F -measure, $Times(ms)$, $Precision$ and $Recall$, respectively. M, L, B, P, F, J, D represent the methods proposed by Mai et al., Bai et al., Prasad et al., Fornaciari et al., Jia et al., DED et al., respectively.

		F	T	P	R			F	T	P	R
Dataset Prasad	M	0.2535	962.2	0.4135	0.2269	Dataset HX	M	0.2573	597.8	0.3109	0.2431
	L	0.0950	1049	0.0700	0.1505		L	0.0534	1560	0.0653	0.0575
	B	0.2395	3470	0.2195	0.2890		B	0.1028	4140	0.0830	0.2181
	P	0.7428	8300	0.8512	0.6813		P	0.3910	3260	0.3364	0.4669
	F	0.3661	12.61	0.8541	0.2330		F	0.3472	19.35	0.8825	0.2161
	J	0.4332	8.42	0.7390	0.3064		J	0.4032	15.69	0.8165	0.2677
	D	0.5549	14.97	0.6437	0.4876		D	0.4467	22.52	0.7836	0.3124
Dataset #1	M	0.2604	1979.5	0.3299	0.2463	Dataset DHX	M	0.2248	1217.04	0.175	0.3737
	L	0.1170	3960	0.1248	0.1415		L	0.0342	1272.04	0.035	0.0621
	B	0.2121	136085	0.2420	0.1909		B	0.0648	48962.1	0.0369	0.4256
	P	0.4512	17130	0.4425	0.4610		P	0.8519	2272.0	0.8823	0.8217
	F	0.5716	16.55	0.7117	0.4777		F	0.4093	40.37	0.3585	0.4769
	J	0.5733	12.61	0.6161	0.5361		J	0.3377	34.68	0.2475	0.5314
	D	0.6378	20.83	0.8312	0.5174		D	0.8906	43.30	0.9347	0.8505

Here, we discuss the competitive advantages of the proposed method against other methods. A curve is split into arcs based on arc curvatures such that our method presents a good effectiveness in detecting small, occluded, and overlapping ellipses. Indeed, even though the arc extraction and arc grouping cost additional computations, they are effective in reducing the computations in the subsequent steps such as the arc grouping and arc clustering. Consequently, the detection process can still be real-time. The reasons regarding false detections are also mentioned. When ellipses are heavily obscured, it may result in false positives (over-detection) because of irrelevant combinations of arcs belonging to different ellipses. In an arc extraction procedure, it sometimes splits the elliptic curves into arcs that do not contribute significant information for ellipse detection due to the lack of curvature, which may generate false negatives (missing detections).

Figure 3-13: The detection practical examples for Dataset Prasad (the 1st three rows) Dataset #1(the last three rows).

Figure 3-14: The detection practical examples for Datasets HX (the 1st three rows) and DHX (the last three rows).

3.3.4 Robustness Analysis of Ellipse Detection in Real Time

Experiment were conducted to investigate the robustness of our ellipse detection method in a sequence of images captured from a static scene with a cluttered background. In fact, detecting ellipses in a single image does not need such a strict requirement. It is rare for processing static nature images in actual applications. The six state-of-the-art approaches are still unable to achieve the competitive accuracy and robustness requirements of real-time applications for industrial or household tasks.

To investigate the robustness of the ellipse detection method in real time scenarios, we use a camera streaming a video (30Hz/s) from six cylindrical objects placed on a table, as illustrated in Figure 3-15. The video containing 214 image frames and the ellipse detector is applied to all images for analysing the robustness of the method during a period. The statistical data of detection results for all six ellipses in images are shown in Figure 3-16.

As expected, the algorithm is fast and robust to the threshold parameters in detecting the ellipses with a fixed centre and orientation and the detector can reach stable detection in a video stream successfully. The detection successes for the ellipses appearing from the left to the right in the horizontal axis for 214 input images are 213, 214, 214, 214, 214, and 203 times, correspondingly. The standard derivations of four parameters (the centre coordinates, semi-major and semi-minor lengths) for the six detected ellipses rest on the ranges of (0.5596, 1.2051), (0.5857, 2.3750), (1.0899, 2.2231) and (0.7673, 2.7055). The standard derivations of orientations vary from 0.0160 to 0.0849 radian. Moreover, the ratios of the mean values of the estimated ellipses parameters (X_c, Y_c, A, B, O) to the ground truth values for six ellipses are 0.9862, 0.9313, 0.9756, 0.9139 and 0.9723, respectively.

Figure 3-15: Real-time ellipse detection in a sequence of 214 images. A cluttered background with light and dark colours is used to verify the robustness of the presented detector.

As shown in Figure 3-17(A), the number of detected ellipses almost keeps a stable level with the time changing. Figure 3-17(B-E) demonstrate four parameters of one of the detected ellipses with the pink colour in Figure 3-16 as a reference case for analysing the detection robustness. Basically, the coordinates (X_c, Y_c) of ellipse centre and semi-axis lengths (A, B) have fluctuations within small ranges. Specifically, we use the ranges of parameters to investigate how geometric configurations (e.g., the centre coordinates and semi-axis lengths) affect the performance of our detector. One semi-axis varies from 56 to 62, and the other one changes accordingly within the range from 53.8 to 58.2. The centre is fixed as $(1095.782, 667.643)$. The fluctuation ranges of its horizontal coordinate are from 1093.5 to 1097.6. The vertical coordinate varies between 653.6 and 659.4.

Figure 3-16: Statistic results of detecting six ellipses in a video stream.

Figure 3-17: Detailed detection performance in all image frames for one of the objects. (F) shows the execution time for each image.

3.4 Summary

In this chapter, we provide an ellipse detector with sufficient accuracy and speed. This is achieved in the presence of complicated scenes. Segmenting arcs based on a changing amount of arc curvatures and grouping arcs depending on the arc convexity-concavity and mutual positions result not only in improving the detection effectiveness (F-measure) but also reduces the detection time considerably. The proposed algorithm imposes the constraints in the scenarios where the severe occlusions result in the lack of visibility. Highly occluded ellipses and cannot be detected. This is inaccurate for short arcs, leading to the failures of detecting the ellipses of minor axis with less than around 15 pixels. Moreover, the proposed ellipse detector cannot deal with concentric ellipses with the almost same sizes. Compared with six other recent methods of detecting ellipses, our extensive experiments demonstrate that our algorithm presents significant advantages for detecting ellipses on images with complex backgrounds as for the trade-off between the detection effectiveness and detection speed.

Chapter 4 Visual Perception based on RGB-D Images

As robots are getting employed in unstructured scenarios that are not formally organized or systematized, they are demanded to manipulate objects in highly cluttered or occluded environments. Performing robotic manipulation is an ongoing and critical challenge for robots, for which perception has been the main bottleneck [88].

A common scenario involves picking and placing tasks, where a robot needs to estimate the pose of the object in a cluttered environment, pick up the object, and move it to a designated position, as shown in Figure 4-1. A large number of efforts are invested in tackling the pose estimation problem [69]. Although the associated challenges have been partly handled in previous works, occlusions among multiple objects can make pose estimation very difficult from a single viewpoint, supporting several ambiguous pose hypotheses. This inaccurate estimated pose greatly limits the ability and generality of robots to reliably make decisions, carry out manipulation actions, especially in typical human environments with considerable clutter. Moreover, even with an appropriate image descriptor, few of the approaches can run at very high speed because of large collection of relevance vectors.

Here, we focus on a specific scenario where one single RGB-D image is input to the perception system. Our target is to design a system to detect object class, and obtain 6D poses of textured and texture-less objects accurately, which transforms a 3D coordinate in the target object space to a 3D coordinate in the camera space even in the presence of occlusions. To achieve robustness, we formulate object detection as a classification problem and pose estimation as a regression problem.

We employ Hough forest coupled with contextual information, including context sensitivity and auto-context, to estimate 6D pose reliably despite the possible occlusion in a cluttered environment by means of reservoir sampling and voting regression distribution. The proposed method benefits from this

Figure 4-1: Robot grasping a soya milk carton occluded in a cluttered environment. The result of pose estimation is visualized by the green bounding box.

integration and achieves results that are superior, in terms of detection speed and accuracy, to the recent works.

The processing pipeline of the proposed system is depicted in Figure 4-2. First, a random forest categorizes sampled pixels that cast votes estimating the 3D object pose in Hough space, which makes efficient use of the reservoir sampling and voting regression distribution based-clustering. Such models with multivariate distribution are recorded at the leaf nodes, calculated based on the annotated data that arrived at each leaf. There is a disconnect between the continuous pose estimation by clustering and discrete voting in Hough space. We also propose a two-stage voting strategy in Hough space to bridge this gap. Here, the fewer but representative centres of clusters are chosen to cast in Hough forest and then the non-maximum suppression is applied to obtaining the maxima, providing not only higher performance of pose estimation, but also computationally efficient. In the next step, the context sensitivity is incorporated with a decision tree to obtain high-level contextual information. To this end, the context sensitivity used in decision trees explores the abundance of contextual information at each node by incorporating a new split function before the growing direction of the node is determined. Subsequently,

Figure 4-2: The processing pipeline of our proposed system. (A) The Hough forest with contextual information; (B) the CNN module for refining the pose; the red arrows denote the input and the output for modules A and B; the blue dot indicates the output energy value.

auto-context is integrated into stacked forests for learning an effective and efficient context model. In this work, the context information is integrated into rich image appearance models via learning a series of classifiers for the context algorithm. Hence, there are techniques of two different levels, namely local hierarchy (the decision tree) and holistic hierarchy (the Hough forest), for blending the contextual information into Hough forest. Finally, in terms of pose refinement, we train a CNN to obtain an energy mapping to the best pose. The experimental results of the proposed approach are not only considerably more accurate than several recent baselines on the publicly available datasets, but, in fact, can be used in robotic grasps while significantly bringing down the computational burden needed for practical applications.

We organize the chapter as follows. We introduce the construction of Hough forest integrated with context information for estimating object's 6D pose in Section 4.1, followed by experimental results on public datasets in Section 4.2. The work is summarized in Section 4.3.

4.1 Methodology

In the section, we first introduce how to use a virtual camera for extracting training patches for Hough forest (A) due to the need of a large amount of training data generated by a virtual camera. Then, the details of training and

testing using decision trees is described (B). Sequentially, the context is integrated into the stacked Hough forests (C). Finally, we choose the final pose using CNN (D).

4.1.1 Extracting Sample Patches for Hough Forest

After building 3D models of the textured and texture-less objects to be grasped, we render synthetic training images from several viewpoints uniformly distributed on the synthetic sphere by putting a virtual camera on discrete positions of the sphere surrounding the object (see Figure 4-3). Note that the viewpoints are obtained on a semispherical surface since the sensor view is restricted, as depicted in Figure 4-4. We also extract the depth invariance as prescribed in [133] and discussed next. Depth values are captured within a certain range such as to discard points that are not from the object. A patch has four channels (RGBD) and the patch size is expressed by V in metric units for obtaining depth-invariance in 3D space. The patches with the four channels are quantitated by a vector of size $V \times V \times 4$. For each training image we extract a set of patches and pre-process patches by normalizing their values to the range $[0,1]$. Last the extracted and normalized patches are provided as input to Hough forest.

4.1.2 Construction of Hough Forest

Decision trees in a random forest in Hough space are built based on a standard random forest framework [53].

Figure 4-3: 3D models of the twelve textured and texture-less objects.

Figure 4-4: Sampling the synthetic viewpoints in the upper hemisphere for object patch generation: the black hemisphere represents a virtual camera. The generated data from synthetic viewpoints are enclosed by the red circles.

Here we employ the Hough forest to predict the distribution over object labels. Given an object label and sample patches, a distribution of object pose is predicted. We train these two distributions via the jointing classification-regression Hough forest, where a classification forest is used, the prediction model is a probability mass function over object labels. In terms of a regression

forest used, the probabilistic prediction model is instead a set of votes for each object.

1) Training

This is the supervised learning, i.e., for training, each training sample is annotated with the information including object class label, 3D object position and orientation. Differ from [42, 134], the appearance of the local patch $\mathcal{P}_i$ where $\mathcal{P}_i$ is the appearance of the 3D local patch (e.g. of $V \times V \times 4$) sampled from RGB-D image is composed of several components: $\{\mathcal{P}_i = (\mathcal{J}_i, c_i, \theta_i, s_i, d_i)\}$. i is the location of the centre pixel of the patch $\mathcal{P}_i$. $\mathcal{J}_i = (I_i^0, I_i^1, I_i^2, \dots, I_i^F, \dots)$ are extracted multiple features at a patch $\mathcal{P}_i$ and I_i^F is the F^{th} feature channel at the patch $\mathcal{P}_i$. In our system, the feature channels, such as depth [135], colour in LAB space, the first and second order gradients in x and y dimensions for the intensity space, LBP, and HoG, are used. The object class is represented by c_i . The vector $\theta_i = \{\theta^x, \theta^y, \theta^z, \theta^{yaw}, \theta^{pitch}, \theta^{roll}\}$ includes six parameters of the object's pose associated to every patch. The components θ^x, θ^y , and θ^z denote position parameters from the point in a RGB-D camera falling on the centre of the training patch to the object position, while $\theta^{yaw}, \theta^{pitch}$ and θ^{roll} are the object rotation angles indicating the object orientation. The binary class label (0 for background patch and 1 for object patch) is denoted by s_i and d_i is the offset from the centroid of the object to the centre of the sample patch. Note that we do not define d_i when image pixels do not belong to an object class.

Each non-leaf node of a tree is assigned a binary test in relation to the patch appearance during training. The training sample is passed to the left or right node depending on the following binary test. The binary test $V_{B,R_1,R_2,\tau}(\mathcal{J})$ is defined as

$$V_{B,R_1,R_2,\tau}(\mathcal{J}) = \begin{cases} 0(left), \frac{1}{R_1} \sum_{i \in R_1} \mathcal{J}_i^g - \frac{1}{R_2} \sum_{i \in R_2} \mathcal{J}_i^g < \tau \\ 1(right), \text{ otherwise,} \end{cases} \quad (4-1)$$

where $\mathcal{J}_i^g$ represents the feature channel g , R_1 and R_2 denote two rectangles within the patch boundaries, and τ is a threshold. The difference between the average values of two rectangular areas rather than single pixel differences is used in testing, which is less sensitive to noise. A novel split function that is introduced next is also employed to do the binary test.

For training, at each non-leaf node, a pool of the binary tests is created with random values of R_1, R_2 and g . At a no-leaf node, we calculate the local maximum value ρ_{max} and minimum value ρ_{min} of all the binary tests. For each test at each node of the current depth, we can get the threshold $\tau(t)$ of the binary test t as follows,

$$\tau(t) = k(\rho_{max} - \rho_{min}) + \rho_{min}, \quad (4-2)$$

where k represents a random value between 0 and 1 and t denotes the order number of binary tests.

By means of randomly determining the uncertainties for splitting, nodes decreasing both the object class and 6D pose are interleaved passing through the decision tree. Thus, patches recorded at the leaf nodes incline to have low variation for the object class label and the pose, and vote with low uncertainties in Hough space. The best binary test can utilize the split function to separate the patches in order to minimize the uncertainty of the object class and 6D pose.

Depending on the classification forest to determine the object class label, the ideal binary test is the one with the minimal entropy stored in a patch in all the tests at each node as follows,

$$U_1(A) = -|A| \cdot \sum P_c \ln(P_c) \quad (4-3)$$

where $|A|$ is the number of patches in a set A and P_c is the proportion of patches with the object class label c in a set A .

Each leaf node does not record the object class but also the object pose. Traversing the binary tree costs much computational resource for training samples while training. To keep memory consumption reasonable, the training samples that arrive at the leaf nodes are summarised by the reservoir sampling

[136] to obtain a fixed-size unbiased training samples. The effect of the reservoir capacity on estimation accuracy of object pose is discussed later. The mean shift [130] is applied to clustering training samples after the reservoir sampling. The detail algorithm is shown in Table 4-1. According to the regression forest, we model the object 6D pose θ at each node as realizations of a random variable with a multivariate Gaussian distribution [137], *i.e.*, $p(\theta) = \mathcal{N}(\theta; \tilde{\theta}, \Gamma)$. $\tilde{\theta}$ is the mean of 6D pose θ and Γ represents the full covariance matrix for each mode.

$$U_2(A) = \ln(|\Gamma(P)|) - \sum_{j \in \{\mathcal{L}, \mathcal{R}\}} \varphi_j \ln(|\Gamma_j(P_i)|), \quad (4-4)$$

where $\mathcal{L}$ and $\mathcal{R}$ represent the left and the right, respectively; P_i is the set of patches reaching node j and P is the set of patches at the parent node of j ; φ_j is the ratio of the number of patches in the node j to the number of patches in its parent node, *i.e.*, $\varphi_j = \frac{P_i}{P}$. Maximizing Eq.(4-4) encourages minimizing the determinant of the covariance matrix Γ , thus decreasing the uncertainty for casting the votes by each patch cluster. The covariance matrix $\Gamma = \text{diag}(\Gamma^\alpha, \Gamma^\beta)$ is block-diagonal, Γ^α and Γ^β denote the covariance matrix among the location vectors and among the rotation angle vectors, respectively. We obtain the following equation,

$$U_2(A) = \ln(|\Gamma^\alpha| + |\Gamma^\beta|) - \sum_{j \in \{\mathcal{L}, \mathcal{R}\}} \varphi_j \ln(|\Gamma_j^\alpha| + |\Gamma_j^\beta|). \quad (4-5)$$

Once the termination criteria are reached at a leaf node, the information of the remaining patches, such as object class label and object pose, is recorded; otherwise, another binary test is used to split the sample patches again, as shown in Figure 4-5. Each leaf node saves the mean of position and orientation vectors that arrived at it, together with their covariance

2) Testing in Hough Forest

Given the trained tree structure above, we describe the approach to classifying the object class label and regressing the object's pose briefly. When a new, unseen RGB-D image of an object is input, patches with the same size as the ones used for training are extracted and pre-processed by mean normalization

for speed up the processing speed, to pass downward through each of the trained trees in a random forest. At the non-leaf nodes of each decision tree, the binary tests guide an input sample patch towards a leaf node. A patch that reached a leaf node l provides an estimation for the parameters of the object's pose in terms of the stored distribution $p(\theta|l) = \mathcal{N}(\theta; \tilde{\theta}, \Gamma)$.

Note that the object class is first determined independently in Hough space where we accumulate the votes of the leaf nodes. The votes can be weighted based on the probability of the associated object class recorded in the leaf node. For the trained trees in a random forest $\mathcal{F}$, a patch $\hat{\mathcal{P}}_l$ arrives in the leaf nodes, the probability $p(c|\mathcal{F}, \hat{\mathcal{P}}_l)$ of object class c is calculated by averaging the probabilities of object class labels at the reached leaf nodes $L_T(\hat{\mathcal{P}}_l)$ that have been stored during the period of training as follows,

$$p(c|\mathcal{F}, \hat{\mathcal{P}}_l) = \frac{1}{N} \sum_{T=1}^N p(c|L_T(\hat{\mathcal{P}}_l)), \quad (4-6)$$

$$p(c|L_T(\hat{\mathcal{P}}_l)) = \frac{1}{m} \sum_{l=1}^m P_c^{L(l)}(T), \quad (4-7)$$

where $p(c|L_T(\hat{\mathcal{P}}_l))$ is defined as the probability of object class c in the reached leaf nodes of the trained tree T in $\mathcal{F}$ including N trees. $P_c^{L(l)}(T)$ denotes the proportion of patches belonging to an object class label c arriving at the leaf node l of the tree T and m represents the total number of leaf nodes recording the patch $\hat{\mathcal{P}}_l$ in the tree T .

To accelerate the voting progress, we adopt the two-stage scheme, which cooperates the continuous probabilistic distribution based on Gaussian distribution and the discretized voting in Hough space together. At each leaf node l , a sample patch provides a pose estimation for the stored distribution $p(\theta|l) = \mathcal{N}(\theta; \tilde{\theta}, \Gamma)$, i.e., a continuous regression output. The variables are summed to produce a Gaussian; its mean is the ultimate estimate of the output

Figure 4-5: The relation of numbers of the nodes at the current depth and the tree depth for 2,4,6,8 decision trees. Each broken line represents a tree.

Table 4-1: The detail algorithm of regression models.

Algorithm: Regression models to summarize the distributions of training samples at leaf nodes

- 1: // Collect training samples arriving at leaf nodes
 - 2: **For all** training samples **do**
 - 3: Calculate the binary test
 - 4: Determine training sample to the left or right
 - 5: Descend tree to reach leaf node l
 - 6: Store training samples with reservoir sampling
 - 7: // Cluster
 - 8: **For all** leaf nodes **do**
 - 9: **For all** the training samples **do**
 - 10: Cluster the training samples using mean shift
 - 11: Store the mean of elements in pose vector
-

parameters and its covariance evaluates the estimation uncertainty based on mean shift interactions.

We fit the Gaussian Mixture Model to the object pose for patches arriving at a leaf during training. The GMM is fitted using mean-shift.

$$p(\theta|c, l) = \sum_{(\vartheta, \tilde{\theta}, \Gamma) \in \mathcal{M}} \vartheta \mathcal{N}(\theta; \tilde{\theta}, \Gamma) \quad (4-8)$$

Where $\mathcal{M}$ is the set of mixture components including the support weight ϑ , the mean $\tilde{\theta}$ and the full covariance matrix Γ . Because the leaves with a high variance are not very informative and add mainly noise to the estimate, we discard the modes with weight ϑ less than an empiric threshold of the highest weight in $\mathcal{M}$.

The order of Hough voting for obtaining 6D pose is first (θ^x, θ^y) , then θ^z , followed by $(\theta^{yaw}, \theta^{pitch})$ and lastly θ^{roll} . Since the use of large votes results in a prohibitive testing time, we must instead utilize a compact representation of the distribution. To speed up, we propose the voting regression model by incorporating a clustering pre-processing step to reduce votes per node. The modes of votes are found by mean shift and sizes of distributions are employed as weights while accumulating votes. We represent the distribution using a set of the cluster centres of the G largest modes found by mean shift. We assign a confidence weight to each chosen cluster according to the size of its cluster, whereby the set of absolute votes cast by all the training samples for the object pose is aggregated using mean shift. Our experiments later highlight the importance of this. Specifically, the mean of all votes returned by the forest is used for the initialization of the mean-shift, which is regarded as the true object position, where most of the votes usually cluster. Such cluster centres are voted in Hough space. The next step is to extract maxima in the smoothed Hough space using non-maximum suppression (NMS). To explore θ^z , we scale the distance towards the pixel to account for the projected size of the object in the image for each object position recorded in the leaf node that has been reached by the patch $\hat{\mathcal{P}}_l$ (see Figure 4-6). Similarly, the orientation votes are clustered as a set of centres to be voted in Hough space. at the leaf nodes in each trained tree.

Figure 4-6: The concept of depth to scale mapping.

Figure 4-7: Detection for object class and pose by casting votes. A,B,C,D represent the voting spaces for object class and the pose including the position and the orientation, respectively. NMS means non-maximum suppression.

Only such variables are further casted to Hough space for determining the object pose, which significantly decreases the computational cost. At each leaf node, for a pose of object class c , the first two values of the Hough voting are

the yaw and pitch angles, respectively. From this Hough space, we can get the points with the largest number of Hough votes. Exploring the zone around each such point, all the leaf nodes in which the patch $\hat{\mathcal{P}}_i$ arrived are obtained. For each leaf node of each object class, the third value of the Hough voting is the roll angle, as illustrated in Figure 4-7. For outliers, auto context (discussed in the next subsection) is integrated into Hough forest for reducing outliers. In subsequent experiments, eliminating outliers is verified to be important for

improving the detection accuracy. Once a full 6D pose is detected, the result of pose estimation is visualized by the bounding box, and the algorithm renders the mesh model of 3D object and adjust the patch dimension to be proportional to the dimension of the bounding box, as shown in Figure 4-8.

4.1.3 Contextual Information for Stacked Hough Forest

We summarize how the contextual information including context sensitivity in each tree and auto-context in stacked forests is integrated into the Hough forest algorithm while learning.

1) The integration of context sensitivity in each tree

In this work, the spirit of context-sensitive decision tree is to use split functions endowed with the capability of integrating contextual information for image pixels at each node, before determining where to route the node. In our work, we do not only consider posterior classification from the previous level of the classifier but also give the regression information since it is applied to random forests in Hough space based on the posterior probability.

Inspired by [15], we impose the idea of prioritized node training to increase the learning effectiveness in a Hough forest. But differently, we do not use loss function to determine the order of a priority queue. This is because the two uncertainties that generate the almost same effect have been used in training and the use of loss function can decrease the learning efficiency at each split node. The new split function is parametrized by a context-sensitive decision tree generating the contextual information in the form of two displacement vectors o_1 and o_2 relative to a location i , a threshold τ that are used to perform a binary test for coupling the contextual sensitivity. The split function that takes a patch $\mathcal{P}_i$ as input is provided as follows,

$$\mathcal{W}(\mathcal{P}_i|o_1, o_2, \tau) = \begin{cases} 0(left), & \text{if } E_{(s,d,s',d') \sim Q_1, Q_2}(\mathcal{Z}) < \tau; \\ 1(right), & \text{otherwise.} \end{cases} \quad (4-9)$$

$$\mathcal{Z} = \begin{cases} K_\sigma(d - d'), & (s, s') = (1, 1); \\ 0, & (s, s') \neq (1, 1). \end{cases} \quad (4-10)$$

Figure 4-8: Renders of textured or texture-less 3D mesh models(red) and the results of pose estimation visualized by the bounding boxes.

where $E_{\mathcal{X} \sim Q}(f(\mathcal{X}))$ represents the expectation of $f(\mathcal{X})$ concerning $\mathcal{X}$ sampled based on a distribution Q that denotes the posterior probability of vote elements for patches arriving at the leaf node. The associated posterior probability of each element in the output space is determined by seeking the probability distribution Q parametrizing the leaf node reached by the patch $\mathcal{P}_i$ routed along the decision tree T . $K_\sigma(\cdot) = \exp\left(-\frac{\|\cdot\|^2}{\sigma^2}\right)$ with σ -algebra [138] is just a Gaussian (or radial basis function kernel). The bandwidth σ is a free parameter controlling the "window" of the kernel and set to 7 in this work. $Q_k = Q(\cdot | (i + o_k), T)$, $k \in \{1, 2\}$, are the posterior probabilities computed from the decision tree T at positions $(i + o_1, i + o_2)$ of the patch $\mathcal{P}_i$, respectively. For Hough forest, the novel split function $\mathcal{W}(\mathcal{P}_i | o_1, o_2, \tau)$ tries to separate the training set into cases with the similar voting direction and length, which indeed reduces the uncertainty uniformly from root to leaf nodes and leads to more stable contextual information.

When a binary test meets the split functions in Eq.(4-1) and Eq.(4-9,4-10), the patch can further be passed to the next node. We choose the split functions randomly. Building upon the split function, the designed split function performs regression and clustering for training samples with similar voting direction and length, instead of performing each task separately.

2) Auto-context coupled among Hough forests

We do not only train a single Hough forest but also a stack of Hough forests $\{\mathcal{F}^1, \dots, \mathcal{F}^\gamma, \dots\}$ in the cascaded way. The whole output of the forest is taken as input to the next forest including the object class label probability $p(c | \mathcal{F}^\gamma, \mathcal{P}_i)$ and estimated 6D pose prediction $p(\theta | \mathcal{F}^\gamma, \mathcal{P}_i)$ in the γ^{th} Hough forest.

Outliers have a critical influence on accumulators in Hough space, leading in inaccurate results. To relieve this issue, it is required to smooth the predictions from the forest $\mathcal{F}^\gamma$ for the outputs of neighbouring pixels before passing to the next forest $\mathcal{F}^{\gamma+1}$. Motivated by [139], but differ from them, we deal with smoothing prediction in the stage of training Hough forest. Calculating parameters of prediction probability in Hough space is similar to clustering

problem [42]. In terms of clustering, the stability has also critical effect on the prediction results [140], which are different from the conclusions made by [139]. It is well known that L1-loss function is more robust and generally unaffected by outliers. However, the instability property of L1-loss means that the accumulators may fluctuate significantly with a small change of input data. In contrast, L2-loss is stable. When the presence of outliers occurs, we remove the undesired outliers in input predictions by means of L1-loss and use a L2-loss function to get a better pose estimation because of the decrease of the number of points. This leads to the improvement of accuracy in seeking the maximum value for Hough votes regarding 6D pose and the optimization of computational resource. Therefore, we combine L1-loss and L2-loss functions together by means of linear functions to discard undesired outliers and improve clustering stability during iterations.

We explain our process of combining L1-loss and L2-loss functions. A median filter is applied to the local neighbourhood of each pixel in an image. Under L1-loss and L2-loss, the median-smoothed object probabilities are defined as

$$h_c^\gamma(c|\mathcal{P}_i, L_1) = \operatorname{argmin}_{i \in \mathcal{N}_i} \|p(c|\mathcal{F}^\gamma, \mathcal{P}_i) - \tilde{p}_i\|_1, \quad (4-11)$$

$$h_c^\gamma(c|\mathcal{P}_i, L_2) = \operatorname{argmin}_{i \in \mathcal{N}_i} \|p(c|\mathcal{F}^\gamma, \mathcal{P}_i) - \tilde{p}_i\|_2, \quad (4-12)$$

where we denote by $\mathcal{N}_i$ a small neighbourhood around the centre pixel in $\mathcal{P}_i$. $\tilde{p}_i$ is the median object probability for each local neighbourhood.

Similarly, we use the method above to smooth the 6D pose prediction $p(\theta|\mathcal{F}^\gamma, \mathcal{P}_i)$, which is crucial for predicting object 6D pose. The median filter is not only robust to outliers but also stable for clustering based on L1-loss and L2-loss. Since the median filter is just used in smoothing data with one dimensionality, the optimum under L1-loss has to be formulated in Euclidean space. Thus, the regularized object 6D pose output is defined as follows,

$$h_\theta^\gamma(\theta|\mathcal{P}_i, L_1) = \operatorname{argmin}_{i \in \mathcal{N}_i} \sum_{T \in \mathcal{F}^\gamma} \|\mu_{i,T} - \theta\|_2, \quad (4-13)$$

$$h_\theta^\gamma(\theta|\mathcal{P}_i, L_2) = \operatorname{argmin}_{i \in \mathcal{N}_i} \sum_{T \in \mathcal{F}^\gamma} \|(\mu_{i,T} - \theta)^2\|_2, \quad (4-14)$$

Figure 4-9: The output of each forest in a stack of forests with auto-context.

where $\mu_{i,T}$ represents the mean value with the highest mixture weight of the pose distribution $p(\theta|\mathcal{F}^\gamma, c, \mathcal{P}_i)$ for the object class c in the tree T from the forest $\mathcal{F}^\gamma$.

To discard undesired outliers and improve clustering stability during iterations, we combine functions of object class label and object pose together by means of the linear function. The final median-smoothed probability is provided as

$$h_c^\gamma(c|\mathcal{P}_i) = \lambda_1 h_c^\gamma(c|\mathcal{P}_i, L_1) + \lambda_2 h_c^\gamma(c|\mathcal{P}_i, L_2), \quad (4-15)$$

$$h_\theta^\gamma(\theta|\mathcal{P}_i) = \lambda_1 h_\theta^\gamma(\theta|\mathcal{P}_i, L_1) + \lambda_2 h_\theta^\gamma(\theta|\mathcal{P}_i, L_2), \quad (4-16)$$

where we denote by λ_1 and λ_2 the weights of $h_c^\gamma(c|\mathcal{P}_i, L_1)$, $h_\theta^\gamma(\theta|\mathcal{P}_i, L_1)$ and $h_c^\gamma(c|\mathcal{P}_i, L_2)$, $h_\theta^\gamma(\theta|\mathcal{P}_i, L_2)$, respectively. We set $\lambda_1 = \lambda_2 = 0.5$. The definition of a feature for this output, to be used in forest $\mathcal{F}^{\gamma+1}$, is straight forward, $f_c(x) = h_c^\gamma(c|\mathcal{P}_i, i + o)$. The location i of the centre of $\mathcal{P}_i$ and the relative offset o , the smooth object class c are included the parameter x . Correspondingly, we define the feature $f_\theta(x) = h_\theta^\gamma(\theta|\mathcal{P}_i, i + o)$ on the smooth object pose output. The parameter x consists of the object class c , the pose θ , the relative offset o and the location i . The set of passed features can be enhanced by updating $f_c(x)$ and $f_\theta(x)$ at the current forest, as shown in Figure 4-9.

4.1.4 Choosing the best pose of object

Although we efficiently have achieved preliminary poses for objects present in the image in the above stage of our pipeline, we still refine them further to

improve the accuracy of estimating the object's pose in the third stage of our pipeline by means of a supervised CNN used in comparing the observed and rendered images. A Gibbs distribution with a CNN as energy function is employed to model the posterior distribution of the pose [52]. Gibbs distribution can be used to model a posterior distribution of the pose H .

$$\delta(H|\Omega, \omega) = \frac{\exp(-E(H, \Omega; \omega))}{\int \exp(-E(H', \Omega; \omega)) dH'} \quad (4-17)$$

where $E(H, \Omega; \omega)$ represents the energy function that maps from a pose H and the observed images Ω to a value, parameterized by the vector ω . Note that the vector ω of learnable parameters consists of CNN weights. The CAD model of the objects are available. To this end, we first render the object in a pose H to get rendered images $\mathbb{R}(H)$ using OpenGL. The constructed CNN can compare Ω with $\mathbb{R}(H)$ to output an energy value $f(\Omega, \mathbb{R}(H); \omega)$. Here the energy function is defined as follows,

$$E(H, \Omega; \omega) = f(\Omega, \mathbb{R}(H); \omega). \quad (4-18)$$

As stated above, we evaluate the quality of a pose H by the energy value from the CNN output. We then choose the best pose among the refined ones according to the minimum CNN energy. This is the final estimate. For comparison, we construct a CNN architecture in which we jointly feed rendered and observed images as separate input channels into the CNN to generate an energy value. Thus, it is an energy value in terms of the CNN output. We first pre-process the original CNN input by cropping the central area with the pose H and resizing it, as suggested by [39]. As shown in Figure 4-10, we employ in total ten input channels to our network, i.e., one rendered depth channel (1) that contains the normalized z component of the object position from the pose H ; one rendered object RGB channels (3) of the object; one probability channel (1) depicting the object probability from Hough forest; one object RGB channels (3) cropped from RGB image; one observed depth channel (1) from the input depth image; one depth mask channel (1) describing whether a depth value is measured in the pixel or not.

Figure 4-10: The CNN architecture.

We train the designed network according to the maximum likelihood paradigm [39, 52]. A convnet model, which is shown in Figure 4-10, is used for computing the energy. C_i , P_i and FC represent the i -th convolutional layer, the i -th pooling layer and fully connected layer, respectively; for $a \times b \times b$, a and b denote the number of filters and the kernel size, correspondingly. The architecture of the designed CNN is illustrated as $\{ Conv(20,3,1) - ReLU - P(2,2) - Conv(40,3,1) - ReLU - P(2,2) - Conv(80,3,1) - ReLU - P(2,2) - Conv(160,3,2) - ReLU - P(2,2) - F(500) - ReLU - F(500) - ReLU - f(\Omega, \mathbb{R}(H); \omega) \}$, where $Conv(a, b, r)$ denotes a convolutional layer with a filters of the size $b \times b$, and stride s . We denote by $P(b, r)$ the max pooling layer of size $b \times b$ and stride r . We use $ReLU$ (rectified linear unit) as activation function in all the convolutional and fully connected layers, together with max-pooling. $F(n)$ indicates fully connected linear layer with n output units. The last output is a value. We train the CNN model constructed from scratch (i.e., randomly initialized weights) using SGD (stochastic gradient descent) with 0.9 momentum and initial learning rate of 0.01 decreased in each epoch to avoid local minima, gradient clipping, and a batch size of 256. In fact, the gradient clipping is useful for training as even robust regression losses generate unstable gradients in the initial period of training. Later we discuss the benefit from the use of the CNN in refining the pose.

4.2 Evaluations and Discussions of Estimation Performance

Here we present the determination of the parameters in terms of the proposed method (4.2.1); self-comparison to show the performance of the method (4.2.2); performance evaluations of pose estimation of multiple objects and occluded object on two publicly available datasets of [38, 39], against the several state-of-art works (4.2.3).

Here we explain three general evaluation metrics. In the experiments for public datasets, the metric defined in [32] and is employed to determine whether the estimated pose is accepted or not. This metric is named by Metric 1 in this work. For an object model M , given the ground truth rotation R_{gr} and translation T_{gr} , the estimated rotation R_{es} and translation T_{es} , the matching score is defined as

$$q = \text{avg}_{x \in M} \|(R_{gr}x + T_{gr}) - (R_{es}x + T_{es})\| \quad (4-19)$$

If $\varepsilon D > q$, the pose estimation is regarded as a correct one, where ε is a constant and D is the diameter of the object. As for symmetric object that have a set of ambiguous views, the matching score is define as follows:

$$\varepsilon = \text{avg} \min_{x_1 \in M, x_2 \in M} \|(R_{gr}x_1 + T_{gr}) - (R_{es}x_2 + T_{es})\| \quad (4-20)$$

We set $\varepsilon = 0.15$ for all the tests.

There is another test metric for evaluating the correctness of pose estimation. If the translational error and angular error is less than 5cm and 5° respectively in comparison with the pose of the ground truth [141], the estimated pose is deemed correct. Such metric is called as Metric 2.

We also use *F-measure* as a performance criterion. It is the harmonic mean of precision and recall. Such evaluation metric can be considered reliable measure of the algorithm's performance since it considers both the precision and recall.

4.2.1 Framework Parameters

We optimize the parameters on a validation dataset created by a virtual camera above. For each 3D mesh model of the object, Hough Forest is trained by varying parameters while fixing other parameters.

The size V of a patch has an important effect on the detection accuracy because a large patch tends to consider a holistic matching, which is sensitive to clutter and occlusions. However, if the size is very small, it may be regarded as noise. With the patch size increasing, F-measure significantly increases. Note that the decrease of F-measure occurs for patches with size equal to 20 and higher because of clutters or occlusions. Figure 4-11(A) shows six different configurations between the patch size and the number of trees, which indicates the remarkably small F-measure variability for the number (4,6,8) of the trained trees when the patch size is 20 and larger. Evidently, a forest with more than four trees imposes the computational burden. In this work, we set the number of trees to 4 and the size V of the patch is 20.

Regardless of using either 50k or 100k training images, the depth of trees is of importance in detection performance, as shown in Figure 4-11(B). Of all the training parameters, the depth appears to generate the most significant effect as it directly impacts the model capacity of the classifier [43]. Figure 4-11(B) shows that F-measure rapidly improves as the tree depth increases, though it starts to level off around depth 32. For 50k images, the over fitting occurs at about depth 28, but it is avoided using the enlarged 100K training images. The deeper tree comes with a large extra memory penalty. Here the maximum depth of trees is set as 32 and the number of training images is 100k.

The effect of the number of forests on the detection performance is shown in Figure 4-11(C), where it is obvious that the detection accuracy increases gently when the number of Hough forests is larger than 3 while the execution time consistently increases. Thus, the best number of random forest is set to 3.

Figure 4-11: Parameter determination and time comparisons. For (F), Wt (without reservoir sampling and distribution compression), RS (Reservoir sampling), DC (Distribution compression).

From Figure 4-11(D), the size of the reservoir has little effect on the detection accuracy. From 200 to 300, a small increase occurs in F-measure. This indicates that even a small set of samples is sufficient to characterize their distribution well for clustering. To quantify the role of the number G of cluster centers, we test our system varying the number of cluster centers. Increasing the number of cluster centers causes a small rise decreases from 0.889 to 0.902, as shown in Figure 4-11(E). We set this number of cluster centers 100.

Figure 4-11(F) shows the effect with and without reservoir sampling and the distributions to represent votes by clustering on the execution time during a testing period. It reveals that the use of these two methods can improve the detection efficiency significantly. In our unoptimized implementation, we train three Hough forests, each forest with 4 decision trees, for 12 objects with 100k images, which costs approximately 10 hours on a personal computer with 8 cores. It spent the vast majority of time on traversing the decision trees; the use of reservoir sampling and voting regression model ensure that the training time totals about 20 minutes.

4.2.2 Performance Evaluation by Self-comparison.

As illustrated in Figure 4-12, the proposed algorithm based on Hough forest with context-sensitivity and auto-context delivers a relative accurate performance on the two metrics for the dataset of [32] where the object is known in advance. On the whole, the integration of the context information into Hough forest evidently improves the performance of 6D pose estimation. The use of auto-context further improves the accuracy than the use of context sensitivity alone, improving by 3.4% and 2.7% for the two metrics. Without context sensitivity and auto-context, the presented estimator fails to estimate the correct poses reliably. For the metric in [32], our approach is very close to 100%. In contrast, since the metric in [141] is more rigorous than the metric in [32] for rotation evaluation, the presented method still achieves the accuracy of 87.1%, which is considerably more than one without contextual information (-18.2%), one with just context sensitivity (-9.7%) and one with just auto-context (-7.0%). Also, the combination of L1-loss and L2-loss boosts the detection performance. Using the combination of L1-loss and L2-loss, compared to L1-loss only or just L2-loss only, results in improvements of 6.2% or 6.8% for the metric in [32] and 3.9% or 5.4% for the metric[141], respectively.

Indeed, the refinement stage is optional, and without it, the estimator already has got a good performance. For the dataset[32], the use of CNN improves the performance of the final 6D pose (-2.1% and -2.4% without the refinement for two metrics[32, 141], respectively), as shown in Figure 4-13. Thus, the excellent performance of our estimator is attributed to Hough forest with contextual information and a novel CNN architecture of the pose choosing. Note that Metric 1 and Metric 2 are proposed in [32] and [141], respectively.

Figure 4-12: The average results for self-comparison. CS (Contextual sensitivity), AC (Auto-context), L1 (L1-loss), L2 (L2-loss).

Figure 4-13: The comparison results for the methods with CNN and without CNN.

4.2.3 Performance Evaluation of Two Public Datasets

1) Evaluation on the dataset of [38]

To illustrate that our approach estimates the poses of multiple objects in the image, we evaluate our algorithm on the dataset of [38], which consists of six sequences and contains the same multiple objects per test image, with significant clutter and different levels of mild occlusion. The proposed estimator is evaluated against ones of Tejani et al. [38], Doumanoglou et al. [55], Drost et al. [142]. The comparison of the F-measure of several methods for this dataset are presented in Table 4-2. Some examples of pose estimation are illustrated in Figure 4-14. The detection results of poses are visualized by the bounding boxes represented by the frames.

The challenge of estimating poses of “Camera” and “Milk box” instances stems from the main factor that these objects are texture-less and of uniform colour in clutter environments. In Table 4-2, our approach is superior to the state of arts on the Camera that has the similar appearance with the clutter and is partially occluded. This also shows the soundness of our method because of being robust to the clutter scenarios. The proposed method can deal with various object appearances with excellent performance such as “Milk box” instances. The “Milk box” object is symmetric and has almost uniform colours so that it becomes very difficult to estimate pose by local detectors, even so our predictors still provides a good performance the compared with other estimators because of the advantages of integrating the contextual information with Hough forest. The target of method proposed by [32] is to estimate the pose of one object per image, thus it does not show a good performance on the dataset of [38]. All in all, our method outperform others in estimating poses of multiple objects. We obtain an average F-measure of 0.906 in the dataset of Tejani, which exceeds the state-of-the-art estimator of Doumanoglou et al. by 10.3%.

Table 4-2: Results of F-measure.

Object	Method			
	Tejani et al.	Doumanoglou et al.	Drost et al	DOP
Camera	37.2%	90.3%	40.7%	91.2%
Coffee cup	87.7%	93.2%	86.7%	97.0%
Joystick	53.4%	92.4%	27.7%	94.6%
Juice carton	87.0%	81.9%	60.4%	90.6%
Milk box	38.5%	51.0%	25.9%	83.0%
Shampoo	75.9%	73.5%	65.1%	87.2%
Average	63.3%	80.3%	51.1%	90.6%

Table 4-3: Results for Metric 1.

Object	Method				
	Hinterstoisser et al.	Krull et al.	Brachmann et al.	Michel et al.	DOP
Ape	81.4%	68.0%	53.1%	80.7%	87.1%
Can	94.7%	87.9%	79.9%	88.5%	91.1%
Cat	55.2%	50.6%	28.2%	57.8%	62.7%
Driller	86.0%	91.2%	82.0%	94.7%	98.5%
Duck	79.7%	64.7%	64.3%	74.4%	79.6%
Eggbox	65.5%	41.5%	9.0%	47.6%	68.6%
Glue	52.1%	65.3%	44.5%	73.8%	67.2%
Puncher	95.5%	92.9%	91.6%	96.3%	97.1%
Average	76.3%	70.3%	56.6%	76.7%	81.5%

Figure 4-14: The examples of pose estimation.

Figure 4-15: The examples of pose estimation.

2) Evaluation on the dataset of [39]

The occlusion dataset created in [39] is an extension of [32], which includes instances of multiple objects occurring serious occlusions. We compare the proposed estimator with the methods of Hinterstoisser et al. [32], Krull et al. [52], Brachmann et al. [39] and Michel et al. [46]. In this case, the estimator is difficult to get the pose of object due to the occlusions. Our method still achieves an average rate of correctly estimated poses of 81.5% using the criterion presented in [32]. Qualitative examples are illustrated in Figure 4-15. The red, green, blue bounding boxes show the detection results of poses of the white glue, toy cat and drilling, respectively.

To evaluate the ability of our approach that estimates the poses of objects occluded, we compare the proposed method against several recent estimators. The results are presented in Table 4-3. Here, our method does with excellent accuracy for estimating pose of “Ape” that has small size, which performs best among these methods. In terms of objects with rich features such as ‘Driller’, our method performs better than the current methods because Hough forest with the integration of context sensitivity enhances the capability that learns a variety of features for Hough forest. For ‘glue’, the performance of our estimator is worse than the cited works. The algorithm proposed in [39] has considerably lower accurate rate compared with other methods since the energy functions based on features designed manually cannot reflect the latent feature information. In comparison with the newest work implemented in [46], our approach realizes an improvement of 4.8%. Note that our method is named as DOP.

4.3 Summary

In this chapter, we presented a new approach to estimating the 6D poses of everyday objects in cluttered environments by means of Hough Forests coupled with contextual information, and refined using convolutional neural network. Our method consistently exceeds state-of-the-art methods on public datasets.

The results indicate that current systems of pose estimation can be improved with suitably learnt features. These results on the pose estimation of objects demonstrate furthermore that our algorithm can be applied not only to the pose estimation of multiple instances present in an image, but also to such scenes where objects may occlude each other and occupy only small fractions of a scene.

Chapter 5 Experiments and Discussions of Robotic Grasp

Since we believe that the most practical way of testing the proposed algorithm of pose estimation is to test it on a physical robot in real applications, we perform an extensive and representative set of robotic grasp experiments on the existing robotic platform. The first series tests track cylindrical objects, and grasp static and dynamic cylindrical objects. Test series two is about grasping single textured or texture-less object with different poses on a table, and grasping three objects that are occluded with each other in cluttered scenes.

5.1 Robotic System and Platform

Two 6-DOF industrial robot arms, namely Universal Robot 10 (UR10) and Universal Robot 5 (UR5), are used to conduct the experiments. Both robots are equipped with a 2-finger Robotiq 85 gripper. UR10 is used in the tracking of cylindrical objects and grasping of static cylindrical objects experiments, and UR5 is used for grasping dynamic cylindrical objects to verify the performance of the pose estimation method, as shown in Figure 5-1. The camera is fixed on a Pan-and-Tilt approximately 1.5 meter above the ground near a table.

The robot arm-UR5 and UR10 have the same repeatability of ± 0.1 mm. They, equipped with a gripper, can reach the maximum distances of 95cm and 140cm, respectively. UR5 and UR10 can support payloads of 5kg and 10kg, correspondingly. Without tactile feedback, the 2-Finger Robotiq 85 gripper has two articulated under-actuated fingers that each have two joints (two links per finger) driven by a single actuator, which automatically adapt to the shape of the object manipulated. Our perception sensor is a RGB-D camera named Microsoft Kinect V1 and Kinect V2 that can provide high quality registered videos and an operation range of 0.8 to 3.5m. The calibration accuracy between Kinect sensor and robot arm base is up to an average of 5mm.

Figure 5-1: The setups of robotic grasping experiments.

All the experiments are executed without code parallelization on a PC with GTX 645, 8GB RAM and an Intel Core i7 processor. We use open-source software such as ROS[143], and Moveit[144] for path planning, and OpenCV[145] for showing 6D pose and Caffe[146] for getting the detection results.

Moreover, an operation range of the linear guide is between 0 to 1m. The control system does not have an encode. The stepper motor rotates a round and the slide on the linear guide moves 75mm.

5.2 Manipulations of Cylindrical Objects

5.2.1 Tracking Cylindrical Objects based on The Proposed Ellipse Detection

We perform the experiment using the proposed detector to further evaluate its performance for robotic manipulation of cylindrical objects. The detector is evaluated in a dynamic scene in which the image sequence includes multiple moving objects with the cylindrical shape. Finally, the method is applied to a robotic task with a dynamic scenario for which the robot needs a capability to track the objects' status for manipulation purposes. Based on practical

Figure 5-2: Snapshots of dynamic detections at 40% of full speed of robotic motion for comparing the presented detector (-D) with Jia's detector (-J) and Fornaciari's detector (a,b,-F) at 0.2Hz sampling frequency.

applications, if the ellipse detection is used in real-time dynamic scenarios, the execution time must be short enough, and the detector must realize a high F-measure.

To evaluate the performance of the presented detector in dynamic scenarios, we construct a set of experiments of moving cylindrical objects at different speeds, and compare our results with two other existing real-time detectors (Fornaciari 2014 and Jia 2017), as shown in

Figure 5-3. In the experiments presented here, we use five objects: four cans with different sizes, and a cup. The objects are randomly placed on a board which is carried with the help of the experiment platform. We use a simple motion model where the manipulator base joint is rotated around 55 degrees at different speeds for illustrating the performance.

While executing the robot motion, the image sequence acquired by the camera is processed in real time by means of the proposed algorithm, and the five parameters of all detected ellipses in each image frame are recorded independently at every time step. Its center coordinates describe the position of ellipses in the image frames. The snapshots of the robot status and our detection results are illustrated in Figure 5-2 at a sampling rate of 0.2Hz. All detected ellipses in all image frames are shown as well in

Figure 5-3 that demonstrates the trajectories of moving objects. The results of the other two ellipse detectors proposed by Jia et al. and Fornaciari et al. are depicted at the second and third rows of

Figure 5-3, respectively, which are applied to the same conditions such as the motion speed and rotation angle. These two detectors cannot discard false ellipses as they always merge close ellipses, which result in excessive fault detections. Regarding quantitative evaluation, F-measure, Precision, and Recall of the presented method are 0.7873, 0.9571 and 0.6814, respectively. However, the detectors proposed by Fornaciari and Jia have worse performances with low F-measure (0.2934, 0.3130), Precision (0.3412, 0.3845) and Recall (0.2573, 0.2639), respectively.

To explore the effect of different moving speeds on the detection performance, eight experiments with various angular velocities of base motion between [0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0] of the full speed (180 degrees/sec) of the robotic arm are performed. As seen in Table 5-1 for experimental results of 8 trials, we achieve stable values of F-measure, Precision, and Recall. The presented detector can keep a high detection effectiveness, yielding more than 75.01% F-measure for the different moving speeds. According to the given experimental

Figure 5-3: Sequence of detected ellipses in a dynamic scenario with multiple moving cylindrical objects. The

Table 5-1: Comparison results for various speeds.

	<i>F-measure</i>	<i>Precision</i>	<i>Recall</i>
1.0 of full speed	0.7885	0.9192	0.7110
0.9 of full speed	0.7707	0.9388	0.6715
0.8 of full speed	0.7797	0.9153	0.6930
0.7 of full speed	0.7506	0.9265	0.6520
0.6 of full speed	0.7873	0.9571	0.6814
0.5 of full speed	0.7842	0.9599	0.6753
0.4 of full speed	0.7501	0.8425	0.6760
0.3 of full speed	0.8154	0.9474	0.7277

results, the motion speed of objects does not have a significant effect on the performance of the ellipse detector.

Failures come in two categories based on the experimental results. The first cause of the failures is due to the cluttered background. The presented ellipse detector sometimes cannot distinguish “real” elliptic edges and “false” edges from the background. However, the presented method still can realize an acceptable performance in dynamic scenarios. The faults happen rarely, and we observe four out of 19 failures. Another cause of failures is when the object is far away from the camera. As a result, the ellipses become smaller gradually until the detector cannot succeed with such small ellipses.

5.2.2 Stacking Static Cylindrical Objects

To enable reliable ellipse detection, we use more constraints than other methods for validating the ellipse candidates. The proposed method could be applied to several robotics applications enabling the robot to mainly depend on 2D information for detection for rapid detection and tracking of cylindrical objects. For instance, the ellipse detector could be deployed to track cylindrical objects on conveyor belts for picking and placing, to detect cylindrical mechanical parts from a cluttered set of different shape objects for sorting them, or track circular shafts and circular holes simultaneously for assembling purposes.

Here, we carried out one experiment with different cylindrical objects, to demonstrate and validate the performance of our approach in a robotic manipulation task. Let us consider a human-robot collaboration task, in which the human operator feeds some objects frequently to a working station that is a table. The robot task is to detect the arrival of new cylindrical objects on the right side of the table and subsequently stack them on the other half side of the table on the left. Thus, the robot requires fast detection of multiple cylindrical objects on the table. It should be also aware of changes in the scene to identify the new objects fed by the operator, to update the position of the latest objects stacked on the left side and to track the status of an object once it moves from the right side to the left.

We perform this task through deploying the proposed ellipse detector. Thus, all the cylindrical objects are identified and tracked by detecting the top of the objects as ellipses. Additional information is needed to capture the 3D coordinate of the detected object to be manipulated. A Microsoft Kinect is applied to obtaining two types of information of RGB and depth. The centroid location of the detected ellipses by RGB information is provided as 2D pixel coordinates. The 2D pixel coordinates of the centroid have to be converted to 3D world coordinates so that the robotic manipulator can move to that location and grasp the cylindrical object. The depth image, which is an encoded grayscale image with each gray value measuring distance to the Kinect camera, is used to obtain the 3D world coordinates of the detected centroid. The 3D

Figure 5-4: Stacking cans in a dynamic scenario. Place the first can (1-A,B,C), stack the second can on the first can (2-A,B,C) and stack the third can on the second can (3-A,B,C).

coordinates of the centroid are then acquired from the point cloud by mapping the 2D pixels coordinate to the registered depth in the point cloud.

Once a new ellipse is detected in the right side of the table, Figure 5-4(1-A)(2-A)(3-A), the 3D position of the top of the object is located and the manipulator is automatically triggered to pick up the object.

Then by real- time detection of ellipses, the manipulated object is tracked, Figure 5-4(1-B)(2-B)(3-B), and its state from a new detected object (ellipses are shown in green) is changed to a stacking object (ellipses are shown in red), once the ellipse moves to the left side of the table. Finally, as the object is placed on the table or stacked on the top of another object, as show in Figure 5-4(1-C)(2-C)(3-C), the targeted position for stacking the next object is updated by getting the 3D position of the latest ellipse on the left side.

The results of ellipse detection and the state of robot arm at the picking instances are given in Figure 5-4, respectively. The results indicate fast and reliable detection and positioning of objects with cylindrical shapes for robot manipulation purposes, such as picking, stacking and placing. The images indicate the detected objects using RGB images captured by the Kinect (V1) camera, which show the states of the robot arm and gripper at manipulating instances.

5.2.3 Grasping of Dynamic Cylindrical Objects

The robotic grasp system consists of three main steps such as perception, inference and manipulation. In dynamic grasp experiments, we first use the proposed detector to detect the ellipse and then, given the position where object is placed by the depth information, the robot infers the next position of the moving object on the linear slide. The robot moves to the next position to wait for the object to be grasped. Last, the robot adjusts the direction of gripper to manipulate the object to the destination and release it.

Here we do not use the point cloud to deduce the space coordinate of cylindrical object. The bottleneck is that the depth based on point cloud has the low update rate and is inaccurate due to noise and weak registering characteristic between the RGB image and depth image. Initially, we just obtain many groups of depth information to calculate the average value and map to the z-coordinate. To avoid the use of the point cloud in dynamic scenarios, our alternative method is to employ the 2D RGB images to determine the position of the object on the linear guide. Without rotating on the conveyor, the position of the object on the guide is taken at the two endpoints of the conveyor. Thus, the slope of the object's path on the conveyor belt relative to the robotic base can be obtained. The pixel values of the image when the object is at the two positions is also recorded. With these information, the object pose along the conveyor can be extrapolated, and obtained at a rate of video.

1) Grasps of moving objects with different speeds

In the first set of grasp experiments, our goal is to grasp a can that is placed on the linear guide with four different speeds. After placing a can on the slider, the camera can capture the position of the can. The grasp is determined only by the estimated orientation and 3D location of the object centre. A successful grasp is defined that the robotic gripper lifts an object 20cm above the linear guide and keeps it until the gripper opens. Figure 5-5 shows the successful rates at each step.

Initially, the robot waits at the home position and uses 1 second to sample the speed of object, determining the 7-DOF gripper configuration. Depending on inverse kinematics, the robot plans a suitable path to be near to the object in a “pre-grasp” position which is defined as 10 cm away from the actual grasping configuration along the approach vector (the normal to the palm of the gripper). This is needed because if the gripper approaches in too early or too late, the gripper will push the object on the linear slide. After arriving at the “pre-grasp” position, the gripper waits for the object moving to the front of the gripper. When that happens, the robot will move 2cm ahead of the newest object position, to perform the grasp action. If it is outside the operating range of the conveyor, the robot will wait for the new pre-grasp position from the returning phase. Lastly, the robot moves to the destination and the gripper opens. We perform four sets of experiments with different speeds, at 0.025, 0.050, 0.075, 0.100 m/s, 30 times for each speed. The main factor affecting the time between picks is most likely due to the operation range of the conveyor. Specifically, when the planned pre-grasp position is beyond the operation range of the conveyor, the robot has to wait for detecting ellipse and sampling the speed of the can in the returning phase. Moreover, the grasp time can still be shorter. Because the robot, UR5, will vibrate in high-speed motion. Here we have to set a speed limit.

The success rate for each trial at various speeds is summarized in Table 5-2. The robot successfully grasped 118 times out of 120 tries, realizing an overall success rate of 98.33%. Specifically, 3 out of 4 experimental sets were

Figure 5-5: The robot grasping cans on the conveyor with 4 speeds in a dynamic scenario. The target is enclosed by a red frame. The speeds of A,B,C,D are 2.5cm,5.0cm, 7.5cm, 10.0cm,respectively. Several detection examples for different speeds in the middle column.

Table 5-2: Experimental results of robotic grasps in dynamic scenarios

Speed (cm/s)	Trials	Successes (single-way)	Successes (Double-way)	Failures	Success rate	Time(s)
2.5	30	30	0	0	100%	0.89
5.0	30	29	1	0	100%	0.51
7.5	30	30	0	0	100%	0.47
10.0	30	25	3	2	93.3%	0.35
Results	120/t	114/t	4/t	2	98.33%/a	0.47/a

/t and /a mean the total value and the average value, respectively.

performed without a single failure. Successful grasps indicate the proposed detector is quite robust and can be used in dynamic scenarios. Table 5-2 also shows two cases on which the grasps failed. For these two failures, the main factor contributed to the failure: the object was touched or knocked down by one finger when the gripper approached to it. The high speed of the can resulted in the inaccurate path planning in the fourth set of experiments. In practical grasp experiments, this robotic gripper is not equipped with any force or tactile feedback so that even a slight error in perception and path planning could lead to a grasp failure. In terms of realizing robotic dynamic grasps, we have to integrate each main module such as visual perception, motion controlling as motion planning suitably. Moreover, we believe that the perception strategies with force or tactile feedback would complement the grasping system and contribute to building a more robust grasping system. However, the results are quite inspiring and provide credible evidence that the proposed detector can be applicable to real scenarios. Several grasp experimental cases are shown in Figure 5-5.

2) Dynamic picking and placing cans

In the second experiment, the robot is used to pick cans from the linear slide objects and place them in a box. Our system first infers the position of the grasped object by ellipse detection and then plans a path to manipulate it on a table. The work scene is set: 10 cans are placed densely on two borders of table. Two persons put cans on the slider randomly. Once detecting the position of object by the proposed detector, the robot will manipulate it to the placed destinations until the end. For each object the robot just attempts to grasp one time. If the position of the grasped object is inaccurate, the robot just performs a grasp one time. The algorithm stream for detecting cylindrical objects will continue seeking the next object so that the robot executes next grasp action.

We evaluate grasps with cases where the ellipse detection is successful by allowing the robot system to execute a grasp and move the object to the destination. We declare failure if the robot does not achieve a grasp or drops the object during or after lifting. Since the maximum opening size is just bigger

Figure 5-6: The steps of robot picking and placing cans. The target is enclosed by a red frame.

than that of the can, small errors in position estimation have a significant effect on a grasp outcome. This is not surprising since the rounded and curved shape made it prone to rolling out of the gripper as it closed.

The speed of the linear guide is fixed to 5.0cm/s. In each trial, the robot picks up the can from the slider and place it in the designated area. Similarly, the system uses 1 second to analyze the position of the can to be grasped. Given an estimated position by the system, the robot moves to the pre-grasp position and wait for the object coming to the front of the gripper. Similar to the experiments above, the planning time and execution time for the robot must match the time

from obtaining the position of ellipse by ellipse detector to the position where the object arrives in the front of gripper. If this position is found to be outside the operation range of the linear guide, the robot will grasp the can in the returning phase. After obtaining the object, the robot places the can in a box near the table.

We ran 30 picking and placing experiments using 10 cans. For the 28 rounds of experiments, we obtained an 93.3% grasp success rate (2 grasp failures out of 30 grasp attempts). Of the two failures, they were due to the estimation of inaccurate positions, resulting in collisions of the fingers with the object when the gripper attempts to grasp. Figure 5-6 shows several screenshots of our robot picking and placing the objects.

5.3 Experiments and Discussion for Novel Picking and Placing Strategy

The robotic grasp system consists of two main steps, namely perception and manipulation. In robot grasp experiments, we first use the Kinect sensor to capture the RGB-D image of the object and the scene. Then the proposed method is used to obtain the best pose of tested object. Finally, given the position where object is placed, the robot adjusts the direction of gripper to grasp the object, list it, move it to the destination and release it.

5.3.1 Robotic Grasp of Textured or Texture-less Objects

In the first set of grasp experiments, our goal is to grasp 12 objects that are placed in five different poses (i.e., varying positions and orientations) for each object in a 40x40cm area. To avoid the collision between the gripper and the object in object dimensions, at least two length dimensions of each of these 12 objects are less than the total opening distance of the gripper. A successful grasp is concluded if the robotic gripper lifts an object 15 cm above the supporting plane and keeps it lifted for 5 seconds.

We first place an object at a random pose on a table in front of the robot. The camera fixed on the Pan-and-Tilt then takes a single image (along with the

depth map) of the object, which is used as the input to our pose estimation algorithm, and determines the 7-DOF gripper configuration.

Depending on inverse kinematics, the robot plans a suitable path to arrive close to the object in a “pre-grasp” position, which is defined as 10 cm away from the actual grasping configuration along the approach direction (the normal of end-effector of robotic arm). Then the gripper moves along the approach direction to the actual grasp position, grasps the object, and lifts it to a position 15 cm higher. A successful grasp depends on three key steps: prediction, reaching and the final grasp.

The first row of Figure 5-7 shows the accuracies at each step. We performed five experiments per object to assess the repeatability. Several grasp experimental cases are shown in Figure 5-7 (the second row). The success rate for each object is summarized in Table 5-3. The robot successfully grasped 56 objects out of 60 trials, realizing an overall success rate of 93.3%. Specifically, 8 out of 12 tested objects were grasped 5 times without a single failure. Successful grasps indicate that our approach can be used in real world repeatably. We found that our algorithm was quite robust to the objects with multiple poses in real scenarios. Even though the objects differ significantly in their appearance and overall shape, the proposed method can obtain an accurate pose.

Table 5-3 also shows some failure cases of our algorithm. As for the 4 grasp failures, the reasons are varied. In tests of the LUX bar Soap, the robot failed to arrive at the pre-grasp position. because the camera obtained partial information only of object, which leads to an inaccurate pose. Of other three failures, for some objects with relative big size and with irregular shape such as the soya milk carton and the shincup respectively, small errors in pose estimation had a significant effect of outcome on grasp. This is not surprising since in stable horizontal poses, the width of soya milk carton is close to the width of the gripper opening. In terms of the shincup, the rounded and curved shape made it prone to rolling out of the gripper as it closed. For the other one failure, it was that a collision of the fingers of the gripper with the object before the grasp

Figure 5-7: The steps of robot grasping an object (the first row) and robot grasping 12 different textured and texture-less objects in cluttered environments (the second and third rows). The target is enclosed by a yellow frame. The object pose is visualized by the green box representing 6D pose segmentation.

Table 5-3: Experimental results of robotic grasps

Object	Trials	Pre-grasp correct	Success times	Success rate
Soya milk carton	5	100%	4	80%
Canned tomato	5	100%	5	100%
Toy duck ¹	5	100%	5	100%
Tooth-paste carton	5	100%	5	100%
Textured-less model	5	100%	5	100%
Potato chip canister ²	5	100%	4	80%
LUX bar Soap	5	80%	4	100%
OREO cookie box	5	100%	5	100%
Canned Lettuce	5	100%	5	100%
Shincup	5	100%	4	80%
Honey lemon drink	5	100%	5	100%
Canned sardine	5	100%	5	100%
Results	60/t	96.67%/a	56/t	93.3%/a

The items with 1,2 are texture-less objects. /t and /a mean the total value and the average value, respectively.

occurred because the high of potato chip canister is very large when the gripper moves towards the potato chip canister.

5.3.2 Grasping Occluded Objects in Cluttered Environments

In the second experiment, the robot is used to move many objects that are occluded by each other in cluttered scenarios. Our robot first infers the 6D pose of the grasped object and then plans a path to place it on the other half of table. We evaluate grasping with cases where pose estimation is successful by allowing the robot system to execute a grasp and move the object to the

Figure 5-8: Robot grasping a soya milk carton (green 6D pose segmentation in the top row), a chip box (blue 6D pose segmentation in the top and middle rows), and a lemon box (red 6D pose segmentation) occluded in a cluttered environment. The target object is enclosed by the yellow frame in each row.

Table 5-3: Experimental results of robotic picking and placing objects.

Object	Trials	Pre-grasp correct	Success times	Success rate
Soya milk carton	10	100%	10	100%
Potato chip canister	10	100%	9	90%
Honey lemon drink	10	100%	8	80%
Results	30/t	100%/a	27/t	90.0%/a

/t and /a mean the total value and the average value, respectively.

destination. We declare failure if the robot does not achieve a grasp or drops the object during or after lifting.

The work scene is set: three grasped objects were placed at random poses densely on half of table in a cluttered environment, which results in the occlusions of different degrees among objects. Since the object was far from its target location, the robot moved to the object, picked it up, moved to the target area and then released it. The scene before and after releasing the object is shown in Figure 5-8. Once detection of the 6D pose of object to be grasped is performed, the robot manipulates it to the target destinations until the end. For each object, the robot attempts to grasp it just one time. If the pose of the grasped object is not correct, the robot attempts the grasp one time. The

algorithm stream for recognizing objects will continue seeking the next object so that the robot executes next grasp action.

We ran grasp experiments in the cluttered environment, described above. First, three objects were selected from a set of 12. In each grasp experiment, the robot had to pick up one object out of the three selected objects and place it in the designated area. There were 10 grasp trials per object, i.e. total 30 trials. The results of these experiments are presented in Table 5-3. Here we obtained an 90.0% grasp success rate (3 grasp failures out of 30 grasp attempts). In general, two failures for the potato chip canister and the lemon box were due to incomplete capture of poses which resulted in the fingertip colliding with the grasped object. The main factor contributing to the failures were that the object was touched or knocked down by one finger when the gripper approached it. Grasp failures occurred when the rolling direction of bounding box showing 6D pose is not perpendicular to the approach direction of the gripper because the pose estimated by our method was not accurate enough. Figure 5-8 shows several screenshots of our robot placing the objects.

In practical grasp experiments, our robotic gripper does not have any force or tactile feedback so that even a slight error in perception could result in a grasp failure. Many objects in clutter environments were only partially seen or not seen at all for the perception sensor, resulting in wrong pose estimation. We believe that methods based on force or tactile feedback can complement our perception algorithm, and help in building a more robust grasping system. Moreover, the grasp success rate can be improved if the gripper has another camera mounted on it, improving the visibility of the object to the gripper.

5.4 Summary

In this chapter, the system devices for experiments such as tracking cylindrical objects, grasping static and dynamic cylindrical objects, grasping dynamic objects, as well as picking and placing moving objects are discussed. The principles and brief steps of all the experiment are presented. We adopt

statistics to analyze the outcomes of experiments and give the corresponding discussions.

Chapter 6 Geometric Design Optimization of Under-actuated Gripper

The robotic grippers are required to be all-purpose and capable such as to perform stable grasps and manipulations in unstructured environments. High cost and high complexity design cannot enable robotic grippers to be refined into products. To moderate these limitations, researchers have proposed under-actuated robotic grippers with tendon-driven mechanisms (TDMs) [108, 147-151]. The use of TDMS can reduce the cost and simplify the structure design due to the less need of actuators, but achieves almost the same actions with full actuated mechanism in some scenarios. Moreover, two-finger robotic grippers with single actuator are prevalent in research topics and industrial applications because of effective grasping capabilities, the mechanical simplicity and low cost [99, 152]. However, robotic gripper design is a very complicated process involving modelling with many parameters [153]. The manipulation performance of an under-actuated gripper significantly depends on the design rather than the control method[154, 155]. Thus, it is essential for presenting a novel versatile optimization design for two-finger grippers with TDMs so that the optimized gripper can realizing stable grasps in household and office environments.

The gripper is also a critical part for realizing stable grasps in the whole grasping system. A stable grasp is that the grasped object can withstand a range of disturbance from external forces or torques and keep the static equilibrium state [148]. It is well-known that performing a stable grasp is the most important target of gripper design. In order to achieve this target, an under-actuated tendon-driven gripper has to be optimized in the design phase. Specifically, a stable grasp should not result in an ejection phenomenon illustrated by [156]. Since all the joints driven by an active tendon are coupling dependently for each robotic finger, the ratio of the contact forces exerted on an object cannot be changed during the period of run-time. A robotic gripper may eject a grasped object and lead to a roll-back phenomenon on condition that an under-actuated robotic hand is designed incorrectly [157]. Moreover, if the

grasped object can withstand a big range of disturbance from the external force/torque, it means the gripper can generate enough forces and torques to resist these disturbances as this equilibrium state is described by the resultant force and torque functions. The grasping stability is considered as a pre-condition to optimize the dimensions of grippers for the gripper to manipulate objects with a big size range. Indeed, the dimension of a gripper has an important effect on the contact force distribution [112].

A tendon route for an under-actuated robotic gripper with TDMs must be designed and optimized carefully such that forces performed on an object are controlled and the resistance of the restoration motion keeps as low as possible. The tendon-route problem is formulated as an optimization problem of the pulley allocation. The optimization of tendon routes can improve the grasping capability.

This work proposes a new practical design approach to optimizing the dimension parameters of an under-actuated robotic gripper and tendon routes based on genetic algorithm by taking account of geometric constraints. Unlike the other related works, this work focuses on design optimization of an under-actuated gripper with TDMs that aims at enhancing the grasping performance for the gripper by adjusting the geometric parameters. The gripper has twenty-two design variables, including three phalange lengths, three phalange widths, three radii of the joint mandrels, a palm width and twelve route variables for allocation of six pulleys. During the course of optimal design, several performance indices are involved, such as the grasping stability, transmission ratio of forces/torques. First, the mathematical model between the active force and the contact forces is extracted to determine the dimension parameters of a gripper. Second, a mathematical model is built to route the tendon with the lowest amount of resistance. Next, two separate optimization processes are presented in detail for the gripper design where three objective functions and multiple geometric constraints are derived and integrated into two fitness models. Finally, the genetic-based optimization algorithm is applied to optimizing the geometric parameters for the stable grasp behavior and to

Figure 6-1: Objects enclosed by bounding boxes (A) and the dimensions of a set of objects approximated by the dimensions of bounding boxes (B).

enhance tendon routes by addressing the layout of pulley allocations. Practical experiments are performed as well to validate the proposed approach.

The rest of the work is organized as follows. Design requirements and variables are proposed in Section 6.1. Section 6.2 builds the models of force/torque transmission between an actuator and fingers and provides the geometric analysis of the tendon routes. Section 6.3 constructs the optimization formulation with respect to the models given by the above section. The detail description of the optimization approach and the optimized results are illustrated in Section 6.4. Section 6.5 describes the practical grasping experiments. Section 6.6 gives a summary for the proposed framework.

6.1 Desired Outcome

6.1.1 Design Requirements

The main design requirements of an under-actuated two-finger gripper with TDMs are set as follows, so as the gripper

- (1) can perform a stable grasp.
- (2) can perform both the enveloping and fingertip grasp.
- (3) includes two 3-phalange fingers with a single actuator.

(4) can return the initial state of the gripper.

(5) can grasp objects with the desired maximum width and a minimum thickness. Considering the robotic application is to manipulate objects, we defined a task of picking items with a maximum width of 100mm.

The first requirement is the most critical condition of realizing a successful grasp. As for the second and third requirements, the enveloping mode is useful for grasping bigger objects as the enveloping grasp can exert the enclosing force in case of ejecting the object. Besides, the gripper with two 3-phalange fingers provides more contact points than a gripper with 2-phalange fingers for the enveloping grasp. Indeed, the fingertip grasp is a pivotal skill for grippers, since it is performed almost as frequently as a force grasp or an enveloping grasp. The gripper cannot exert an enveloping grasp if the object to be grasped is placed on a flat surface. In this case, the gripper uses the fingertips to pick up a small object. To reduce the resistance of restoration motion [111], the tendon routes must be designed and optimized as for the fourth requirement. Considering the robotic application is to manipulate objects, we defined a task of picking items with different shapes and size but with a maximum width of 100mm regarding the fifth requirement. In addition, the size of some daily used objects is explored and diverse objects within the desired size range are selected to be used for performance assessment. The dimensions of a set ($n=40$) of objects, which are used in households and offices [96], are measured approximately by that of the enclosing bounding boxes, as shown in Figure 6-1.

6.1.2 Design Variables

Figure 6-2 shows the schematic view of our gripper design. The robotic gripper includes two 3-phalange fingers driven by tendons. For the design of the initial states of three phalanges, we will adopt the expansion model of the reference [96]; θ_1 represents the angle between the first phalange and the palm, and θ_i is the angle between the phalanges i and $i - 1$, $i = 2, 3$. The initial angles of the

Figure 6-2: A schematic of an under-actuated robotic gripper with TDMs.

three joints are given, such as $\theta_1' = \frac{\pi}{4}$, $\theta_2' = \frac{\pi}{4}$, $\theta_3' = 0$. The fingertip of the third phalange is designed as a curved surface of a quarter of a circle with the

radius $2W_3$ and the two end points of the arc are tangent to the two sides of the third phalange, as shown in Figure 6-2. For the fingertip grasp, the fingertip with the curved surface will be used frequently. Thus, the radius of the designed fingertip is almost equivalent to the length of the third phalanges. Based on the design requirements, the design variables, which comprises of phalange lengths and the pulley allocation around each joint, are $\omega = (L_1, L_2, L_3, r_1, r_2, r_3, W_1, W_2, W_3, L_0, \varphi_{j1}, \varphi_{j2}, m_{j1}, m_{j2})^T$, $j = 1, 2, 3$, where L_1, L_2, L_3 denote the phalange lengths and L_0 is the half width of the palm; r_1, r_2, r_3 denote the radius of the three joints, respectively; W_1, W_2, W_3 are the half widths of the first, second and third phalange. The pulley positions used for guiding the tendons can be illustrated by the polar coordinates φ_{j1}, m_{j1} and φ_{j2}, m_{j2} whose centre is that of the joint mandrel j , as shown in Figure 6-2. Specifically, $\varphi_{j1,2}$ represents the angles with the positive direction pointed by an arrow mark between the centre axis of the phalanx $L_{j,j-1}$ and the contacting

line from the center of joint j to the pulley for guiding a tendon mounted on the phalanx $j, j - 1$, correspondingly; $m_{j1,2}$ denotes the distance between the centre of the joint j and the pulley attached on the phalanx $j, j - 1$, respectively.

6.2 Problem Modelling

The force/torque transmission from the actuator to robotic fingers [110] determines the grasping forces/torques that exert an object. The models of force/torque transmission are built for optimizing the dimensions of robotic fingers, to a lesser extent, to reduce the subjective considerations from the topology of anthropomorphic hand. Moreover, the geometric model for pulley allocations is constructed to optimize tendon routs for relieving the resistance of restoration motion.

6.2.1 The Model of Force/torque Transmission

1) The transmission of tension forces and joint torques

To obtain the relation of the tendon tension force and the joint torque, we can establish the kinematics of tendon-driven mechanism [103]. There exist two tendon-driven types, including the passive tendon with a spring and the active driven by an actuator in Figure 6-3.

Depending on the kinematic relation between the tendon extension $l \in \mathbb{R}^L$ and the variables such as the joint angle vectors θ and the actuator rotation angle vectors η in Figure 6-3, the tendon transmission mechanism can be described as

$$l = l(\theta, \eta, l_0) = \begin{bmatrix} l_a(\theta, \eta) \\ l_p(\theta, l_0) \end{bmatrix}, \quad (6-1)$$

where $l_a(\theta, \eta)$ and $l_p(\theta, l_0)$ are the extensions for active tendon parameters θ, η and passive tendon parameters θ, l_0 , respectively. Specifically, $\theta = (\theta_1, \theta_2, \dots, \theta_N) \in \mathbb{R}^N$ and $\eta = (\eta_1, \eta_2, \dots, \eta_A) \in \mathbb{R}^A$. L, N and A represent the number of tendons, joints and actuators, correspondingly. l_0 denotes the initial extension for the passive tendon. Moreover, setting J_k to be a Jacobian matrix

Figure 6-3: (A) Passive tendon contacted a spring and (B) Active tendon driven by a motor.

that maps the joint angle velocity vector $\dot{\theta}$ to the tendon extension rate $\dot{l}$. The derivative of the tendon extension regarding time can be provided by

$$\dot{l} = \begin{bmatrix} \dot{l}_a(\theta, \eta) \\ \dot{l}_p(\theta, l_0) \end{bmatrix} = J_k \dot{\theta} + \begin{bmatrix} R_a \\ 0 \end{bmatrix} \dot{\eta} \quad (6-2)$$

with $J_k = \begin{bmatrix} J_a \\ J_p \end{bmatrix}$,

where J_a and J_p are the corresponding Jacobian matrixes, and R_a is the diagonal matrix whose diagonal elements are the pulley radius connected with actuators. Then, depending on the principle of virtual work, the next equation is obtained:

$$\tau = -J_k^T F_t. \quad (6-3)$$

Since the tension force is just considered, the negative sign is referred in the above equation. F_t represents the vector of tendon tension force, which is constantly positive since all the tendons cannot generate negative tension force. τ is the joint torque vector.

2) The transmission of contact forces and joint torques

The idlers are applied to compress tendons, which allow the tendons to be in contact with mandrels during the period of joint rotation, such as to ensure the moment arm of the joint torque is the radius of joint. A torque spring used in a practical robotic finger is equivalent to two linear springs for simplifying the model, as shown in Figure 6-4 (A). According to this equivalent model, a finger has seven tendons in the desired design, including one active tendon and six passive ones in Figure 6-4 (B). Referring to [106, 148], all the radii of joints are supposed to be identical in order to simplify the model of the tendon-driven mechanism. However, in this work, the specific radius values are given to calculate the equation of each driven tendon. The radii of joint pulleys are assumed to be r_1, r_2 and r_3 . The mapping Jacobian matrix J_k is obtained based on the following principle. The elements of the mapping Jacobian matrix are determined by the rotational direction of the joint axes and the tendon route. But in fact, the elementary line transformation of the mapping Jacobian matrix has no effect on the function of the mechanism. That is, the mapping Jacobian matrix J_k of elementary transformation is equivalent. Thus, the definition of direction for the joints makes no difference to the tendon-driven mechanism. For instance, the positive direction of rotational joint is defined to be pointing out of the work, the mapping Jacobian matrix J_k is described as follows:

$$J_k^T = \begin{bmatrix} r_1 & r_1 & -r_1 & 0 & 0 & 0 & 0 \\ r_2 & 0 & 0 & r_2 & -r_2 & 0 & 0 \\ r_3 & 0 & 0 & 0 & 0 & r_3 & -r_3 \end{bmatrix} \quad (6-4)$$

The tendon tension force F_t is presented by

$$F_t = \begin{bmatrix} F_{t1} \\ F_{t2} \\ F_{t3} \\ F_{t4} \\ F_{t5} \\ F_{t6} \\ F_{t7} \end{bmatrix} = \begin{bmatrix} F_a \\ k_1 \Delta \theta_1 \\ -k_1 \Delta \theta_1 \\ k_2 \Delta \theta_2 \\ -k_2 \Delta \theta_2 \\ k_3 \Delta \theta_3 \\ -k_3 \Delta \theta_3 \end{bmatrix} \quad \text{with} \quad \begin{bmatrix} \Delta \theta_1 \\ \Delta \theta_2 \\ \Delta \theta_3 \end{bmatrix} = \begin{bmatrix} \theta_1 - \frac{\pi}{4} \\ \theta_2 - \frac{\pi}{4} \\ \theta_3 \end{bmatrix} \quad (6-5)$$

where $F_{ti} (i = 1, 2, \dots, 6, 7)$ represents the tension force of each tendon. The assumption is that the stiffness of two springs attached on each joint is identical. $k_i (i = 1, 2, 3)$ are the stiffness of linear springs for the joint 1, 2, and

Figure 6-4: Schematic diagram of the tendon-driven mechanism regarding an under-actuated finger. The purple circle on the right represents an actuator.

3, respectively. F_a indicates the tension force of the active tendon and $\Delta\theta_i$ ($i = 1, 2, 3$) are the variations of joint angles with respect to the initial values $\frac{\pi}{4}, \frac{\pi}{4}, 0$ for θ_1, θ_2 and θ_3 , correspondingly. From Equation (6-3), we can obtain the following equation:

$$\tau = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = - \begin{bmatrix} F_a r_1 + 2k_1 \Delta\theta_1 r_1 \\ F_a r_2 + 2k_2 \Delta\theta_2 r_2 \\ F_a r_3 + 2k_3 \Delta\theta_3 r_3 \end{bmatrix}. \quad (6-6)$$

3) The model of the contact forces

The active forces, which are generated by the actuator and transmitted by the phalanges, will be used in the optimization procedure. In practical grasping applications, sometimes the palm of the gripper also provides a supporting force to the grasped object. However, a supporting force belongs to a passive force and will not be optimized.

The presented gripper is a planar mechanism with a single actuator. Because of the symmetry, we only focus on a half of this mechanism that is made up of three phalanges and three joints. As illustrated in Figure 6-5, L_i is the length of the phalange i ; P_i denotes the distance from the joint i to the contact point; W_i indicates the half width of the phalange i ; the contact force is F_i for the joint i ; τ_i is the torque of the joint i , $i = 1, 2, 3$. According to the principle of virtual work, the equilibrium equation of the i^{th} phalanx can be provided by

$$\tau = J_i^T F_i \quad (6-7)$$

where J_i is Jacobian matrix of the finger with three phalanges, and F_i denotes the force of the i^{th} contact point regarding the fixed palm frame of reference; τ is the i^{th} joint torque of the under-actuated robotic finger. The geometric approach is applied to achieving the Jacobian matrix J_i that maps the contact force F_i ($i = 1, 2, 3$) exerted on a grasped object to the joint torque τ . M_1 , M_2 and M_3 are the moment arms of the torque τ_1, τ_2, τ_3 , respectively. In Figure 6-5 and Figure 6-6, the magnitude of the applied torque τ_3 is presented as:

$$|\tau_3| = |F_3 M_3|, \quad (6-8)$$

However, there are two frequently-used grasping modes for the gripper such as the enveloping grasp and the fingertip grasp. The flat planar part of the third phalange will be blocked by the second phalange when the third joint does the flex motion due to the high ratio of the radius of the curved surface to the length of the third phalange. During the grasping period, the contact point of the third phalange with the grasped object is on the curved surface rather than the flat plane. As illustrated in Figure 6-6, O is the center of the curved surface of the fingertip whose radius is $2W_3$.

When the contact point is on the curved surface in Figure 6-5 and Figure 6-6, the moment arm depending on the geometric analysis is described as follows

$$M_3 = \frac{P_3^2 + \left(\sqrt{(2W_3)^2 - [P_3 - (L_3 - 2W_3)]^2} - W_3 \right)^2}{P_3}.$$

Figure 6-5: The schematic diagram of the geometric model for the enveloping grasp. The red dots represent the contact points.

Figure 6-6: The schematic diagram of geometric model for the fingertip grasp. The red dot represents the contact point.

The torques τ_2 and τ_1 , which are illustrated in Figure 6-5, can be provided as listed below

$$\tau_2 = F_2 M_2 + F_3 (P_3 + L_2 \cos \theta_3), \quad (6-9)$$

$$\tau_1 = F_1 M_1 + F_2 (P_2 + L_1 \cos \theta_2) + F_3 P_A \quad (6-10)$$

with $P_A = P_3 + L_2 \cos \theta_3 + L_1 \cos(\theta_2 + \theta_3)$ and $M_i = \frac{P_i^2 + W_i^2}{P_i}, i = 1, 2$.

Furthermore, the transpose of the Jacobian matrix can be obtained as follows

$$J_i^T = \begin{bmatrix} M_1 & P_2 + L_1 \cos \theta_2 & P_A \\ 0 & M_2 & P_3 + L_2 \cos \theta_3 \\ 0 & 0 & M_3 \end{bmatrix}. \quad (6-11)$$

The phalanges of the gripper are regarded as rigid bodies. Each phalanx does not have more than one contact point with the grasped object. Due to the low speed movement and the light quality of the fingers compared with F_a , the dynamic properties of the gripper will not be analyzed and the grasp maintains the static equilibrium[158]. Thus, combined Equation (6-6) and (6-7), the contact force F_1, F_2, F_3 are described by

$$F_3 = \frac{1}{M_3} (F_a r_3 + 2k_3 \Delta \theta_3 r_3), \quad (6-12)$$

$$F_2 = \frac{1}{M_2} [F_a r_2 + 2k_2 \Delta \theta_2 r_2 - F_3 (P_3 + L_2 \cos \theta_3)], \quad (6-13)$$

$$F_1 = \frac{1}{M_1} (F_a r_1 + 2k_1 \Delta \theta_1 r_1 - F_2 P_B - F_3 P_A) \quad (6-14)$$

with $P_B = P_2 + L_1 \cos \theta_2$.

The spring stiffness is not only sufficiently strong to keep the phalanges return the initial positions overcoming the resistance without the actuator forces but also as low as possible in order to relieve the actuated force consumption. As a standard mechanical part, the stiffness k_i is a constant. Since the actuated force F_a is far larger than the spring tension force F_t , the contact force can be simplified as follows

$$F_3 = \frac{1}{M_3} F_a r_3 \quad (6-15)$$

$$F_2 = \frac{1}{M_2} [F_a r_2 - F_3 (P_3 + L_2 \cos \theta_3)], \quad (6-16)$$

$$F_1 = \frac{1}{M_1} [F_a r_1 - F_2 P_B - F_3 P_A], \quad (6-17)$$

6.2.2 The Geometric Model for Pulley Allocations

1) Force transmission efficiency for a tendon through joints

Due to the friction, the tendon tension force cannot be transmitted totally when the tendon wraps the joint mandrel. The enveloping angle is defined as the central angle that is made up of the center of the joint and two radii the two end tangent points formed a tendon wrapping a joint mandrel, as shown in Figure 6-7. In Figure 6-7, O is the center of the curved surface of the fingertip whose radius is $2W_3$. Let an enveloping angle α approach an infinitesimal angle θ that is the equilibrium point. The tangent and normal equilibrium equations are provided as follows

$$(F_{tan} + dF_{tan}) \sin \frac{d\theta}{2} + F_{tan} \sin \frac{d\theta}{2} = dN, \quad (6-18)$$

$$(F_{tan} + dF_{tan}) \cos \frac{d\theta}{2} - F_{tan} \cos \frac{d\theta}{2} = \mu dN \quad (6-19)$$

where N represents the supporting force from the object compressed against the surface of the joint mandrel, while μ is the friction coefficient between a tendon and the object. Then, as θ is a tiny value, $\sin \frac{d\theta}{2} \approx \frac{d\theta}{2} \approx 0$, $\cos \frac{d\theta}{2} \approx 1$. Substituting these two equations into (6-18) and (6-19), we can obtain the following equations:

$$F_{tan} d\theta = dN, \quad (6-20)$$

$$dF_{tan} = \mu dN. \quad (6-21)$$

Thus, the function equation can be expressed as follows:

$$\frac{dF_{tan}}{F_{tan}} = \mu d\theta. \quad (6-22)$$

Integration of both sides yields

$$\int_{F_{resist}}^{F_{tension}} \frac{dF_{tan}}{F_{tan}} = \int_0^\alpha \mu d\theta. \quad (6-23)$$

Figure 6-7: Schematics of the enveloping angle (A) and forces exerting a surface along the tangent direction (B).

Thus,

$$F_{external} = F_{resist} e^{\mu\alpha}, \quad (6-24)$$

where $F_{external}$ denotes the contact tangent force generated by the active tendon tension force and F_{resist} expresses the resisting force for keeping the current state.

Based on the above derivation, we can come to a conclusion. Specifically, when the friction coefficient μ is a constant value and the resisting force, F_{resist} , is a fixed value, the larger the enveloping angle α is, the more the capability of resisting the tendon tension force, $F_{external}$ is. Finally, the efficiency η of the force transmission for the tendon wrapping the mandrel is

$$\eta = \frac{F_{resist}}{F_{external}} = \frac{1}{e^{\mu\alpha}}. \quad (6-25)$$

2) The geometric model determining the tendon routes

The tendon routes can be guided by the joint mandrel and pulleys. The geometric parameters of determining the tendon routes, $m_{j1}, \varphi_{j1}, m_{j2}, \varphi_{j2}$, ($j = 1, 2, 3$), are illustrated schematically in Figure 6-8. There are three cases for the tendon routes in Figure 6-8 such as (A) tendon routes of the first and second joints: the two phalanges for the first or second joint form the interaction angle

$\frac{3}{4}\pi$; (B) the third joint at the initial position: the distal and middle phalanges configure the angle π ; (C) the state of the maximum flex angle for all the joints which is $\frac{1}{2}\pi$; α_j is the tendon enveloping when the tendon wraps around the joint mandrel j ; the green dashed lines represent the position of the phalange L_j when the phalange L_j rotates relative to the phalange L_{j-1} ; θ_j represents the intersection angle between the phalanges L_j and L_{j-1} . As the capability of the springs is equivalent to that of the passive tendons for a robotic finger, a tendon referred in the latter sections is an active tendon rather than a passive tendon. The tendon is the unique component attached to the motor and controlled at runtime. The joint mandrels and pulleys can lead the tendon to slide through a series of the route points such as to change the direction of the active tension force and determine the joint torques generated by the tendon. The moment arm regarding the joint mandrel equals the radius of the joint mandrel r_j if and only if the tendon is touching the mandrel. Therefore, the moment arm must be first confirmed and then determines the torque in real time. The critical condition, which allows the tendon to touch the joint mandrel all the time during the rotational motion of joint, is that the contacting line of the pulleys on the two sides of the joint mandrel is tangent to the mandrel when the two phalanges form the maximum flex angle $\Delta\theta_{imax} \left(\Delta\theta_i = \theta_i - \frac{\pi}{4} \text{ or } \theta_i, \Delta\theta_{imax} = \frac{\pi}{4} \text{ or } \frac{\pi}{2} \right)$, as shown in Figure 6-8.

Based on the derivations in the above part, the enveloping angle α_j determines the efficiency of the force transmission for the tendon wrapping the mandrel if the coefficient of friction is a constant. When the phalanges can be back to the initial states without the active force, the springs must overcome the resistance from the tendon routes. In order to increase the force transmission efficiency, the enveloping angle α_j at the initial position should decrease suitably under the condition of tendon touching the joint mandrel. Thus, the enveloping angle α_j is provided as:

$$\alpha_j = \varepsilon - \left(\cos^{-1} \frac{r_j}{m_{j1}} - \varphi_{j1} + \cos^{-1} \frac{r_j}{m_{j2}} - \varphi_{j2} \right) \quad (6-26)$$

Figure 6-8: The tendon routes guided by the joint mandrel and pulleys. The blue dots represent the pulleys.

where ε represent the initial intersection angle between the two phalanges. In detail, ε is $\frac{3}{4}\pi$ for the first and second joints and ε is evaluated by $\frac{3}{4}\pi$ regarding the third joint.

6.3 Design Optimization

6.3.1 Objective Functions

1) The objective functions for the dimension optimization

Having the relation between contact forces and dimension parameters, the two criterions of evaluating the grasping stability are described based on the sum of the grasping forces and the distribution of contact forces.

The first objective function (Equation. (6-27)) is written as the sum of grasping forces. A bigger contact force means the grasped object can withstand a stronger external disturbance for improving the grasping stability. The maximization of this objective ensures that the gripper can realize a stable grasp readily.

$$\delta_1 = F_1 + F_2 + F_3 \quad (6-27)$$

The second objective function (Equation. (6-28)) is the magnitude of difference values among the contact forces. There should be not the big difference among the contact forces in case of ejecting an object [156, 157]. The minimization of this objective will ensure that the exerted forces are almost uniform. For describing this metric, we use the function δ_2 to build the expressions as follows

$$\delta_2 = (\bar{F} - F_1)^2 + (\bar{F} - F_2)^2 + (\bar{F} - F_3)^2 \quad (6-28)$$

with $\bar{F} = \frac{F_1 + F_2 + F_3}{3}$

where δ_1 represents the total contact force and δ_2 donates the discrete magnitude for the contact forces. If a phalange does not include a contact point with the grasped object, the force from this phalange will be zero. For example, the second and third phalanges will not exert the forces for the fingertip grasp, that is, F_1 and F_2 are zeros.

2) The objective function for the tendon routes

The object function (Equation. (6-26)) is the enveloping angle for the active tendon wrapping the mandrel. We will allow this objective to be minimum to improve the force transmission efficiency. The object function τ_j is illustrated as follows

$$\tau_j = \alpha_j \quad (6-29)$$

6.3.2 Geometric Constraints

1) Constraints for the dimension optimization

(a) We refer to the human finger and the length ratio among phalanges [159] for allowing the geometric dimensions to be within a suitable range. All the geometric dimensions must fall into the required ranges according to Table 6-1.

Table 6-1: Dimension constraints for robotic gripper

Geometric dimension(mm)	L_1	L_2	L_3	r_1	r_2	r_3	$2W_1$	$2W_2$	$2W_3$
Lower limit(mm)	40	25	15	6	6	6	10	10	10
Upper limit(mm)	60	40	25	15	15	15	20	20	20

(b)The geometric bounds of the rotation angles of the joints are given as follows (in radian),

$$\frac{\pi}{4} \leq \theta_1 \leq \frac{\pi}{2};$$

$$\frac{\pi}{4} \leq \theta_2 \leq \frac{\pi}{2};$$

$$0 \leq \theta_3 \leq \frac{\pi}{2}.$$

(c) The length of the phalanx $L_i (i = 1, 2, 3)$ must be larger than the contact position $P_i (i = 1, 2, 3)$. The function g_i represents the difference between these two parameters. Specifically,

$$g_1 = P_1 - L_1 \leq 0;$$

$$g_2 = P_2 - L_2 \leq 0;$$

$$g_3 = P_3 - L_3 \leq 0.$$

Thus, the formula $\max(0, g_i)$ can be described by

$$\max(0, g_i) = \begin{cases} 0, & g_i \leq 0 \\ g_i, & g_i > 0 \end{cases}.$$

(d) The fingertip is always used for manipulating small objects. The fingertip accounts for the large ratio of all the third phalange in volume [160]. We set the ratio of the radius of the fingertip to the length of the third phalange as a geometric constraint for the optimization procedure.

$$0.7 \leq \frac{2W_3}{L_3} \leq 1.$$

We use the functions γ_1 and γ_2 to describe these constraints as follows

$$\gamma_1 = \begin{cases} 0, & 0.7L_3 - 2W_3 \leq 0 \\ 0.7L_3 - 2W_3, & 0.7L_3 - 2W_3 > 0 \end{cases}$$

$$\gamma_2 = \begin{cases} 0, & 2W_3 - L_3 \leq 0 \\ 2W_3 - L_3, & 2W_3 - L_3 > 0 \end{cases}$$

(e) The distance P_3 between the contact point and the third joint for the third phalanx L_3 must be larger than $(L_3 - 2W_3)$. The following function ϵ represents the difference between P_3 and $(L_3 - 2W_3)$,

$$\epsilon = P_3 - (L_3 - 2W_3).$$

Hence, the constraint formula $\zeta(\epsilon)$ is presented as

$$\zeta(\epsilon) = \begin{cases} \epsilon^2, & \epsilon \leq 0 \\ 0, & \epsilon > 0 \end{cases}$$

(f) The width of the phalange i must be not less than the diameter of the joint i such that the joint mandrel cannot be exposed outside of the fingers. We build two functions, such as $\min(W_1, W_2, W_3)$ and $\max(r_1, r_2, r_3)$, to describe the relation between the width of the phalange and the radius of the joint mandrel. The function $\min(W_1, W_2, W_3)$ obtains the minimum value of W_1, W_2, W_3 and the function $\max(r_1, r_2, r_3)$ gets the maximum value of r_1, r_2, r_3 . The geometric constraints are provided as follows:

$$\min(W_1, W_2, W_3) \geq \max(r_1, r_2, r_3).$$

2) Constraints for the optimization of tendon routes

(a) According to the geometric constraints in Figure 6-8: The tendon routes guided by the joint mandrel and pulleys. The blue dots represent the pulleys., the initial geometric constraints for the tendon touching the joint mandrel can be given by

$$h = \frac{\pi}{2} - \left(\sin^{-1} \frac{r_j}{m_{j1}} + \varphi_{j1} + \sin^{-1} \frac{r_j}{m_{j2}} + \varphi_{j2} \right) \leq 0.$$

(b) Geometric constraints for the allocation parameters of the pulleys are provided by Table 6-2, Table 6-3, Table 6-4. The function $\max(a, b)$ is used in getting the maximum of a and b .

Table 6-2: The first joint: ranges of the parameters

Route parameters	Lower limit	Upper limit
$m_{11}(\text{mm})$	$\max(15, r_1)$	$\max(15, 0.5L_1)$
$\varphi_{11}(\text{rad})$	0	0.25π
$m_{12}(\text{mm})$	$\max(15, r_1)$	$\max(15, 0.5L_0)$
$\varphi_{12}(\text{rad})$	0	0.25π

Table 6-3: The second joint: ranges of the parameters

Route parameters	Lower limit	Upper limit
$m_{21}(\text{mm})$	$\max(10, r_2)$	$\max(15, 0.5L_2)$
$\varphi_{21}(\text{rad})$	0	0.25π
$m_{22}(\text{mm})$	$\max(10, r_2)$	$\max(15, 0.5L_1)$
$\varphi_{22}(\text{rad})$	0	0.25π

Table 6-4: The third joint: ranges of the parameters

Route parameters	Lower limit	Upper limit
$m_{31}(\text{mm})$	$\max(10, r_3)$	$\max(15, L_3)$
$\varphi_{31}(\text{rad})$	0	0.25π
$m_{32}(\text{mm})$	$\max(10, r_3)$	$\max(15, 0.5L_2)$
$\varphi_{32}(\text{rad})$	0	0.25π

6.4 Optimization Procedure and Results

6.4.1 Optimization Method

Since a function with many variables is difficult to isolate each parameter and get the derivation, Genetic Algorithm (GA) [161] is adopted in optimizing the geometric dimensions of a robotic gripper and pulley allocations for guiding the tendon routes.

6.4.2 Fitness Function Constructions

1) Fitness function for the dimension optimization

The first fitness function $f_{dimension}$ is constructed based on the two objective functions under the condition of some constraints. For an optimization method, GA itself has no capability to handle the constraints. When dealing with the

functional optimization with constraints using GA, we can make it be the problem without constraints. The lower and upper limits of the geometric dimensions can be considered during the period of encoding the chromosomes, while a penalty function p is introduced in a fitness function such that the optimization with constraints is transferred to the corresponding optimization that does not contain constraints but the penalty function p , to deal with the equations and inequality constraints. In order to ensure a stable grasp and resist to a large external disturbance, the total force δ_1 tends to be as high as possible. As to the second criteria δ_2 , the contact forces should be well distributed among the phalanges, which come close to be uniform, to avoid large local forces exerted on a grasped object. Therefore, the second criteria δ_2 should be as small as possible, in other words, $\frac{1}{\delta_2}$ leans to be larger. Finally, to achieve the maximum objective value, the fitness function with the geometric constraints is defined by

$$f_{dimension} = \left(\delta_1 + \frac{k}{1+\delta_2} + p \right) \delta_3, \quad (6-30)$$

with $p = c \sum_1^n (\max(0, g_i) + \zeta(\epsilon) + \gamma_1 + \gamma_2)^2$ and $\delta_3 = 1$ or 0 . If the grasp is successful, δ_3 becomes 1 and otherwise 0. Since the actuated force F_a is a constant in the fitness function and different from the reference [110], it has no effect on the change amount of $f_{dimension}$ and will not be optimized. The formula $\frac{1}{\delta_2}$ is amplified k times to compensate the difference of δ_1 and $\frac{1}{\delta_2}$ in the numeral magnitude. In this work, the actuator in the gripper prototype can execute a known force to the active tendon based on the positive transmission mechanism. As the difference of δ_1 and $\frac{1}{\delta_2}$ is about 1000 in numeral magnitude, k can be specified 1000. In terms of the second item in Equation (6-30), the

Figure 6-9: The grasping space of gripper with two 3-link fingers under different angle combinations.

formula $(1 + \delta_2)$ cannot enable the denominator to be zero. c is the penalty coefficient. In general, c is one order of magnitude bigger than $f_{dimension}$ [161] and it is assigned to be 1000. Specifically, satisfying the constraints, the value of the fitness function becomes larger. In contrast, if violating the constraints, the value of the fitness function greatly decreases due to the large penalty coefficient so that the genes cannot pass on the next generation.

2) Fitness function for the optimization of pulley allocation

We can build the second fitness function $f_{routing}$ to explore the minimum fitness value by the object function τ_j with the corresponding constraints. The fitness function $f_{routing}$ with respect to the tendon routes is defined as follows

$$f_{routing} = \alpha_j + ch^2 \quad (6-31)$$

with $c = 1000$.

6.4.3 Optimized Results

The populations are generated by a random method. For ensuring the generality of the solution, we adopt the different number of the initial population such as

20, 25, 30, 35 and 40. In this work, the single-point crossover operator is used for making the genes from the two parents to be exchanged. As for a mutation operator, it can maintain the genetic diversity and the probability of mutation is set 0.01. Since it is difficult to determine the convergence of GA, the general method is to configure the number of evolutionary generation. The number of the evolutionary generations is confirmed to be 100 as the stop criteria experimentally. All the setting GA parameters will be applied to optimizing the dimensions of the gripper and tendon routes.

1) Optimized results of the geometric dimensions

The fitness function includes the second and third joint angle variables. Since the joint angles have different combination configurations for different grasps, these variables do not need to be optimized and can be given constants. The second joint angle (θ_2), which is between $\frac{\pi}{4}$ and $\frac{\pi}{2}$, is sampled every $\frac{\pi}{24}$. While the sampled step increment of the third interaction angle (θ_3) within between 0 and $\frac{\pi}{2}$ is also $\frac{\pi}{24}$. Thus, there are 72 combinations (θ_2, θ_3) of different angles in total. The grasp space is illustrated by different angle combinations in Figure 6-9. For each angle combination (θ_2, θ_3), the optimization procedure is performed and finally, we adopt the average values of optimized parameters of 72 times to be the optimized results, as shown in Table 6-5. We will adopt the approximate values of the optimized results in fabricating the robotic gripper. According to the geometric relation in Figure 6-10. The width of the maximum stretching state d for the gripper can be determined as follows

$$d = 2L_0 + 2L_1 \cos \theta'_1 - 2W_3$$

The maximum desired width of the grasped object w_{obj} ($w_{obj} = 100mm$) must satisfy the following inequation

$$w_{obj} \leq d.$$

Thus, the palm width of the gripper $2L_0$ can be confirmed.

2) Optimized results of tendon routes

Figure 6-10: The initial state of the gripper to grasp an object (A). The gripper grasps the elliptical object (B) and the rectangle object (C).

Table 6-5: Geometric dimensions, average optimized results and final values

Geometric dimensions(mm)	L_1	L_2	L_3	$2r_1$	$2r_2$	$2r_3$	$2W_1$	$2W_2$	$2W_3$
Average optimized results(mm)	54.78	33.81	25.22	15.74	7.73	16.44	17.82	16.97	18.32
Final values(mm) ^a	54.8	33.8	25.2	15.7	7.7	16.4	17.8	17.0	18.3

The final values are adopted in making the gripper.

Figure 6-11: The first joint: the objective values of different generations and the current optimized values.

Figure 6-12: The second joint: the objective values of different generations and the current optimized values.

Figure 6-13: The third joint: the objective values of different generations and the current optimized values.

Table 6-6: The first joint: the optimized results

Route parameter	$m_{11}(\text{mm})$	$\varphi_{11}(\text{rad})$	$m_{12}(\text{mm})$	$\varphi_{12}(\text{rad})$
Optimization parameters	24.797	0.58	17.35	0.052
Final values	24.8	0.6	17.4	0

Table 6-7: The second joint: the optimized results

Route parameter	$m_{21}(\text{mm})$	$\varphi_{21}(\text{rad})$	$m_{22}(\text{mm})$	$\varphi_{22}(\text{rad})$
Optimization parameters	16.557	0.529	24.83	0.426
Final values	16.6	0.5	24.8	0.4

Table 6-8: The third joint: the optimized results

Route parameter	$m_{31}(\text{mm})$	$\varphi_{31}(\text{rad})$	$m_{32}(\text{mm})$	$\varphi_{32}(\text{rad})$
Optimization parameters	16.61	0.025	17.867	0.445
Final values	16.6	0.0	17.9	0.4

The tendency of the fitness values with respect to the different generations is shown in Figure 6-11, Figure 6-12,

Figure 6-13, for the optimization of tendon routes. The optimized results of the geometric parameters of determining the tendon routes, such as $m_{j1}, \varphi_{j1}, m_{j2}, \varphi_{j2}$ previously introduced in Figure 6-8, are given in Table 6-6, Table 6-7 and Table 6-8 for the three joints. Moreover, the hardware fabrication is considered in designing the robotic phalanges and thus, the geometric parameters adopted can be merely approximate values. Table 6-6, Table 6-7 and Table 6-8 also present the final parameters adopted in making the robotic finger, correspondingly.

6.5 Experimental Results

By adopting the final geometric parameters (dimensions and tendon routes) obtained by the proposed method, as shown in Table 6-5, Table 6-6, Table 6-7 and Table 6-8 the gripper design is finalized for manufacturing and a prototype is fabricated accordingly, as shown in Figure 6-14.

Figure 6-15: The practical grasping cases.

The assessment method previously proposed in [100] for a successful grasping is used in our grasping experiments. An object is put in the centre of the workspace. The grasp is considered a success, if the gripper could grip the object and carry it till the object loses contact with the table surface, while the object remains between fingers for at least 10 seconds after the gripper is moved up about 15 cm. The gripper releases the grasped object when the tendon tension force is lost. In our experiment, the gripper was applied to grasping 40 objects, which were found in household and office environments, with the size range from the maximum gripper span to a business card. Experimental results indicate that the gripper can steadily grasp objects with sizes ranging from 8mm (the diameter of a pen) to 102mm (the diameter of a cup). Specifically, the gripper picks up 38 objects successfully while a business card and a coin cannot be grasped with success since the contact position between the fingers and the object is higher than the height of the object. The real application cases are presented in Figure 6-15. The gripper can perform stable enveloping grasps when the first phalanges touch the object to be stopped such that the second and third joints can flex to enclose the object. If the first and second phalanges are not blocked by the object, the gripper can execute a fingertip grasp by the distal phalanges closing in the opposite direction. As well, the gripper can passively adapt to the shape for grasping objects with irregular shapes due to the compliant characteristics of the under-actuated gripper with TDMs.

6.6 Summary

Geometric parameters of a two-finger gripper driven by TDMs are studied in this work by extracting a mathematical model of the force/torque transmission between an active force and contact forces based on the geometric analysis. In addition, a mathematical model is presented to obtain the transmission efficiency of the tension force when a tendon wraps a joint mandrel by the geometric relations. Genetic Algorithm is applied to optimizing the dimension of the gripper and the tendon routes. Specifically, we construct the two genetic-based fitness functions to be optimization objectives based on the grasping stability and the efficiency of the force transmission. The geometrically optimal approach provided by us has the characteristics of the versatility and can also be referred to optimizing most of the under-actuated robotic gripper with TDMs. We have validated this presented method by constructing an under-actuated gripper with TDMs depending on the optimized results and performing the practical experiments. In particular, the gripper can realize the stable grasps for a wide range of objects in household and office environments, and adapts to the shape of the grasped object because of the under-actuated mechanism.

Chapter 7 Conclusion

The focus of this thesis is on the robotic perception and grasp. Three important issues, namely tracking and grasping of static and dynamic cylindrical objects based on the proposed ellipse detection, pose estimations of multiple objects and occluded objects, optimal design of an under-actuated robotic gripper with two 3-link fingers are studied. For each of these topics, some fundamental and experimental results have been obtained.

For the ellipse detection, three main points have been addressed in Chapter 3. In this chapter, we propose a geometric approach based on gradient analysis to detecting ellipses in practical scenarios with a high detection effectiveness and low execution time. The balance of low execution time and detection effectiveness is obtained through (a) effective combination of rejecting curves that do not contribute to ellipse detection, (b) smart grouping of arcs potentially suitable for detecting ellipses, and (c) utilizing pre-computed gradient information powerfully at multiple steps to alleviate computation needs of the conventional approaches that do not exploit gradients directly. Compared with six other recent methods of detecting ellipses stated in Chapter 4 and Chapter 6, our extensive experiments demonstrate that our algorithm presents significant advantages for detecting ellipses on images with complex backgrounds in terms of trade-off between the detection effectiveness and detection time.

For the problem of the pose estimations of objects in cluttered and occluded environments, addressed in Chapter 4, an extremely cost-effective method based on Hough voting is proposed. The proposed approach consists of two key steps, including the pose estimation and the pose choosing. The context sensitivity used in decision trees explores the abundance of contextual information at each node by incorporating a new split function before the growing direction of the node is determined. Moreover, auto-context is integrated into stacked forests for learning an effective and efficient context model. In this work, the auto-context is integrated into rich image appearance

by learning a series of classifiers. For pose refinement, we train a CNN to obtain an energy value that maps to the best pose. Our method consistently exceeds state-of-the-art pose estimators on public datasets. The results indicate that current systems of pose estimation can be improved with suitably learnt features.

In Chapter 5, the robotic experiments are introduced to verify the proposed methods. In our robotic experiments, the robot successfully tracks a cylindrical object to operate in real time, recognizes a cylinder object, blocks and also sorts cylinder objects from ones with different shapes in a cluttered environment. The robotic grasp experiments illustrate that our method can significantly improve view-based pose estimation of objects, which has direct applications in robotic grasp tasks. The results on the pose estimation of objects demonstrate furthermore that our algorithm can be applied not only to the pose estimation of multiple objects, but also to such scenes in which objects may occlude each other and occupy only small fractions of a scene.

Chapter 6 focus on solving the problem of optimal design of a two-finger gripper driven by tendon-driven mechanisms. We build the mathematical model of the force/torque transmission between an active force and contact forces and the mathematical model of the transmission efficiency of the tension force when a tendon wraps a joint mandrel by the geometric relations to analyse the dimension parameters of a gripper. The genetic-based optimization algorithm is applied to optimizing the geometric parameters for the stable grasp behavior and to enhance tendon routes by addressing the layout of pulley allocations. Practical experiments of robotic grasps are performed as well to validate the proposed design approach.

7.1 Contributions

The strengths of this thesis are derived from several main novelties across the description steps, each having a specific contribution. We enumerate and highlight the main contributions of this thesis below.

7.1.1 Tracking and Grasping of Cylindrical Objects based on The Ellipse Detection

1) Ellipse detection

- This thesis presents an ellipse detection method that combines the advantages of arc extraction and arc grouping. Arc detection involves important arc splitting steps, where our propositions help in precise arc detection at low computation expense. Arc grouping using geometric constraints guarantees the effectiveness of ellipse detection and optimizes the computation cost.
- In the step of smooth arc extraction, we propose a new approach of identifying the precise splitting points (sudden changes) in order to achieve better segmentations from curves to smooth arcs that may belong to ellipses. A coarse search for sudden changes is first performed with a big range, and then such points are determined with a finer scope.
- We use just one inequality with angle vectors to confirm splitting points (sudden changes), as compared to more than one inequalities used by other contemporary methods. This provides advantage of computation speed over other methods.
- The direction of vector formed by bounding box enclosing an arc is used for determining a convexity-concavity of arc precisely to classify an arc further. This speeds up the efficiency of grouping of arcs.
- We present a method to estimate the ellipse centre by an iterative mean-shift clustering algorithm, which improves its robustness to noise and obtains a more precise centre comparing the existing methods that determine ellipse centres.

- We adopt the ratio of half of the circumference of the bounding box enclosing an arc and the sum of the semi-axes lengths to measure the integrity of ellipse to improve the detection accuracy. This is significantly faster than other more complicated approaches of measuring the integrity of the ellipse.

2) Tracking of cylindrical objects

Based on the fast ellipse detection, we propose the method of tracking cylindrical objects. The method does not depend on the point cloud by means of 2D image. The localization and tracking are implementable for both static and dynamic environments for different users. To evaluate the performance of the presented detector in dynamic scenarios, we construct a set of eight experiments of moving cylindrical objects at different speeds. Experimental validations of 8 trials of tracking cylindrical objects illustrated that we achieve stable values of F-measure, Precision, and Recall. The presented detector can keep a high detection effectiveness, yielding more than 75.01% F-measure for the different moving speeds.

3) Grasping of cylindrical objects in static and dynamic scenarios

- The proposed method could be applied to several robotics applications enabling the robot to mainly depend on 2D information for detection for rapid detection and tracking of cylindrical objects. The experiment showed the ellipse detector could be deployed to detect cylindrical objects from a cluttered set of different shape objects for stacking them. It should be also aware of changes in the scene to identify the new objects fed by the operator, to update the position of the latest objects stacked on the left side and to track the status of an object.
- We propose a perception approach that can be applied in grasping dynamic cylindrical objects based on the ellipse detection in real time. Here we do not rely on the point cloud to

deduce the space coordinate of cylindrical object. To avoid the use of the point cloud in dynamic scenarios, our alternative method is to employ the 2D RGB images to determine the position of the object on the linear guide. With the information, the object pose along the conveyor can be extrapolated, and obtained at a rate of video. In terms of grasps of moving objects with different speeds, the robot successfully grasped 118 times out of 120 tries, realizing an overall success rate of 98.33%. As for dynamic picking and placing cans in an industrial line, we ran 30 picking and placing experiments using 10 cans and obtained an 93.3% grasp success rate (2 grasp failures out of 30 grasp attempts).

7.1.2 Pose Estimations of Multiple and Occluded Objects based on Learning

1) The proposed method of pose estimation

- We present a method to estimate the ellipse centre by an iterative mean-shift clustering algorithm, which improves its robustness to noise and obtains a more precise centre comparing the existing methods that determine ellipse centres.
- We present an use of the reservoir sampling and voting regression model based-clustering to summarize the object pose distributions at leaf nodes enables training from large data sets and permits a test performance with high speed.
- We make use of context sensitivity to obtain an extended Hough forest for efficient estimation of 6D pose on an RGB-D image. We integrate contextual information at each node of a decision tree by a new split function defined as a predictor.

- We use auto-context for enhancing the learning performance of Hough forest. We propose a scheme to deal with the issues of the outliers and instability during the iterations between forests by combining L1-loss and L2-loss normalizations.
- 2) Grasping objects based on the pose estimation in cluttered and occluded environments

The proposed method in this thesis can be applied to grasping textured and texture-less objects in cluttered and occluded environments. The experimental results illustrate that our approach is not only considerably more accurate than several recent baselines on the publicly available datasets, but, in fact, can be used in robotic grasps. Moreover, our method significantly reduces the computational time needed for practical applications. We implement our algorithm on an industrial robot, and obtain success rates of 93.3% and 90.0% for two sets of grasp experiments on a highly varied set of objects, respectively.

7.1.3 Geometric Design Optimization of An Under-actuated Gripper

- 1) This work proposes a new practical design approach to optimizing the dimension parameters of an under-actuated robotic gripper and tendon routes based on genetic algorithm by taking account of geometric constraints.
- 2) We build the mathematical model between the actuated force and the contact force for an under-actuated tendon-driven robotic gripper based on the geometric analysis for determining the dimension parameters of a gripper.
- 3) A mathematical model is constructed to obtain the transmission efficiency of the tension force when a tendon wraps a joint mandrel by the geometric relations.
- 4) We propose the geometric model of transmission characteristics determined by the tendon routes for reducing the resistance. A tendon

route for an under-actuated robotic gripper with TDMs is designed and optimized carefully such that forces performed on an object are controlled and the resistance of the restoration motion keeps as low as possible. The tendon-route problem is formulated as an optimization problem of the pulley allocation.

- 5) The geometrically optimal approach provided by us has the characteristics of the versatility and can also be referred to optimizing most of the under-actuated robotic gripper with tendon-driven mechanisms. Practical experiments are performed as well to validate the proposed approach.

7.2 Future Work

Future works would update this algorithm to detect even more heavily occluded ellipses in real time. We also consider segmenting cylindrical objects by the presented ellipse detector and edge information based on deep learning to get 3D coordinates, which allows a robot to sort multiple circle and elliptic mechanical components in industrial environments. Moreover, obviously, the proposed method cannot deduce the object orientation because the proposed ellipse detector just uses 2D information. Future works will focus on further improvement on the accuracy of the proposed ellipse detector, as well as incorporating other sensing modalities and complementary perception methods to leverage robotic manipulation in more complicated scenarios. For example, we will use an approach based on deep learning to deduce 3D coordinates and 3D orientations. After getting 6D pose, the robot can plan more advanced manipulation strategies to pick cylindrical objects with various configurations in complex and dynamic industrial environments.

In the current implementation, we manually designed the features of Hough forest for the pose estimation of textured or texture-less objects. Automatically unsupervised feature learning is interesting but also very challenging and left

for future work. As for a future work, we also prefer to explore how to use a considerably stable estimator of 6D pose in robotic manipulations in real time. Moreover, we also explore the effect of different light conditions on the accuracy of 6D pose estimation.

Future works would extend the optimization to the 3-dimension space. The thickness of the palm and phalanges and the fingertip shape will be explored and optimized, which allows an under-actuated tendon-driven robotic gripper to perform a stable grasp.

We would integrate the proposed perception system and the designed under-actuated gripper together to perform the stable grasps of a large scale of objects.

Based on the comprehensive studies we have accomplished, we would focus on how to extract features by the sparse autoencoder, to enhance the learning effectiveness by means of contextual information and to relieve the uncertainty of follow-up machine learning using the projective invariant pruning. We use sparse autoencoders to learn feature vectors in unsupervised feature learning, which avoid manually designing the features for improving learning effectiveness. Context information are iteratively learnt in the field of scene understanding to boost the learning efficiency. We are pursuing the general and reproducible models of contextual information on improving learning effectiveness. We leverage a newly developed projective invariant, which reflects the intrinsic geometry of image appearance to significantly prune the undesired objective candidates and to pick out potential ones. Our aim is to apply novel solving strategies for realizing real-time practical applications on robot.

References

- [1] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, no. 1, pp. 11-15, 1972.
- [2] S.-C. Zhang and Z.-Q. Liu, "A robust, real-time ellipse detector," *Pattern Recognition*, vol. 38, no. 2, pp. 273-287, 2005.
- [3] A. A. Sewisy and F. Leberl, "Detection ellipses by finding lines of symmetry in the images via an hough transform applied to straight lines," *Image and Vision computing*, vol. 19, no. 12, pp. 857-866, 2001.
- [4] N. Guil and E. L. Zapata, "Lower order circle and ellipse Hough transform," *Pattern Recognition*, vol. 30, no. 10, pp. 1729-1744, 1997.
- [5] L. Xu, E. Oja, and P. Kultanen, "A new curve detection method: randomized Hough transform (RHT)," *Pattern recognition letters*, vol. 11, no. 5, pp. 331-338, 1990.
- [6] N. Kiryati, Y. Eldar, and A. M. Bruckstein, "A probabilistic Hough transform," *Pattern recognition*, vol. 24, no. 4, pp. 303-316, 1991.
- [7] H. I. Cakir, B. Benligiray, and C. Topal, "Combining feature-based and model-based approaches for robust ellipse detection," in *Signal Processing Conference (EUSIPCO), 2016 24th European*, Budapest, Hungary, 2016, pp. 2430-2434: IEEE.
- [8] S. Chen, R. Xia, J. Zhao, Y. Chen, and M. Hu, "A hybrid method for ellipse detection in industrial images," *Pattern Recognition*, vol. 68, pp. 82-98, 2017.
- [9] H. Dong, D. K. Prasad, and I.-M. Chen, "Accurate detection of ellipses with false detection control at video rates using a gradient analysis," *Pattern Recognition*, vol. 81, pp. 112-130, 2018.
- [10] R. Grbić, D. Grahovac, and R. Scitovski, "A method for solving the multiple ellipses detection problem," *Pattern Recognition*, vol. 60, pp. 824-834, 2016.
- [11] H. Dong, I.-M. Chen, and D. K. Prasad, "Robust ellipse detection via arc segmentation and classification," in *Image Processing (ICIP), 2017 IEEE International Conference on*, Beijing, China, 2017, pp. 66-70: IEEE.
- [12] F. Mai, Y. Hung, H. Zhong, and W. Sze, "A hierarchical approach for fast and robust ellipse extraction," *Pattern Recognition*, vol. 41, no. 8, pp. 2512-2524, 2008.

- [13] Y.-B. Lin and C.-P. Young, "High-precision bicycle detection on single side-view image based on the geometric relationship," *Pattern Recognition*, vol. 63, pp. 334-354, 2017.
- [14] M. Fornaciari, A. Prati, and R. Cucchiara, "A fast and effective ellipse detector for embedded vision applications," *Pattern Recognition*, vol. 47, no. 11, pp. 3693-3708, 2014.
- [15] D. K. Prasad, M. K. Leung, and S.-Y. Cho, "Edge curvature and convexity based ellipse detection method," *Pattern Recognition*, vol. 45, no. 9, pp. 3204-3221, 2012.
- [16] X. Bai, C. Sun, and F. Zhou, "Splitting touching cells based on concave points and ellipse fitting," *Pattern recognition*, vol. 42, no. 11, pp. 2434-2446, 2009.
- [17] J. Liang, Y. Wang, and X. Zeng, "Robust ellipse fitting via half-quadratic and semidefinite relaxation optimization," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 4276-4286, 2015.
- [18] K. Hahn, S. Jung, Y. Han, and H. Hahn, "A new algorithm for ellipse detection by curve segments," *Pattern Recognition Letters*, vol. 29, no. 13, pp. 1836-1841, 2008.
- [19] H. Dong, I.-M. Chen, and D. K. Prasad, "Robust Ellipse Detection via Duality Principle with a False Determination Control," in *Proceedings of 2nd International Conference on Computer Vision & Image Processing*, 2018, pp. 223-234: Springer.
- [20] Z.-Y. Liu and H. Qiao, "Multiple ellipses detection in noisy environments: A hierarchical approach," *Pattern Recognition*, vol. 42, no. 11, pp. 2421-2433, 2009.
- [21] B.-K. Kwon, Z. Teng, T.-J. Roh, and D.-J. Kang, "Fast ellipse detection based on three point algorithm with edge angle information," *International Journal of Control, Automation, and Systems*, vol. 14, no. 3, p. 804, 2016.
- [22] B. Jian and B. C. Vemuri, "Robust point set registration using gaussian mixture models," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 8, pp. 1633-1645, 2011.
- [23] S. Zafari, T. Eerola, J. Sampo, H. Kälviäinen, and H. Haario, "Segmentation of overlapping elliptical objects in silhouette images," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5942-5952, 2015.

- [24] C. Liu and W. Hu, "Effective method for ellipse extraction and integration for spacecraft images," *Optical Engineering*, vol. 52, no. 5, pp. 321-336, 2013.
- [25] C. Arellano and R. Dahyot, "Robust ellipse detection with Gaussian mixture models," *Pattern Recognition*, vol. 58, pp. 12-26, 2016.
- [26] S. Mulleti and C. S. Seelamantula, "Ellipse Fitting Using the Finite Rate of Innovation Sampling Principle," *IEEE Transactions on Image Processing*, vol. 25, no. 3, pp. 1451-1464, 2016.
- [27] V. Pătrăucean, P. Gurdjos, and R. G. von Gioi, "Joint A Contrario Ellipse and Line Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 788-802, 2017.
- [28] C. Liu and W. Hu, "Ellipse fitting for imaged cross sections of a surface of revolution," *Pattern Recognition*, vol. 48, no. 4, pp. 1440-1454, 2015.
- [29] Q. Jia, X. Fan, Z. Luo, L. Song, and T. Qiu, "A Fast Ellipse Detector Using Projective Invariant Pruning," *IEEE Transactions on Image Processing*, 2017.
- [30] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge, "Comparing images using the Hausdorff distance," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 15, no. 9, pp. 850-863, 1993.
- [31] S. Hinterstoisser *et al.*, "Gradient response maps for real-time detection of textureless objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 5, pp. 876-888, 2012.
- [32] S. Hinterstoisser *et al.*, "Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes," in *Asian conference on computer vision*, Daejeon, South Korea, 2012, pp. 548-562: Springer.
- [33] H. Tjaden, U. Schwanecke, and E. Schömer, "Real-Time Monocular Pose Estimation of 3D Objects using Temporally Consistent Local Color Histograms," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, USA, 2017, pp. 124-132.
- [34] R. Kouskouridas, A. Tejani, A. Doumanoglou, D. Tang, and T.-K. Kim, "Latent-class hough forests for 6 dof object pose estimation," *arXiv preprint arXiv:1602.01464*, 2016.
- [35] S. Hinterstoisser *et al.*, "Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes," in *Computer Vision*

- (ICCV), *2011 IEEE International Conference on*, Barcelona, Spain, 2011, pp. 858-865: IEEE.
- [36] R. Rios-Cabrera and T. Tuytelaars, "Discriminatively trained templates for 3d object detection: A real time scalable approach," in *Proceedings of the IEEE International Conference on Computer Vision*, Portland, USA, 2013, pp. 2048-2055.
 - [37] T. Hodaň, X. Zabulis, M. Lourakis, Š. Obdržálek, and J. Matas, "Detection and fine 3D pose estimation of texture-less objects in RGB-D images," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, Hamburg, Germany, 2015, pp. 4421-4428: IEEE.
 - [38] A. Tejani, D. Tang, R. Kouskouridas, and T.-K. Kim, "Latent-class hough forests for 3D object detection and pose estimation," in *European Conference on Computer Vision*, Zurich, Switzerland, 2014, pp. 462-477: Springer.
 - [39] E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother, "Learning 6d object pose estimation using 3d object coordinates," in *European conference on computer vision*, 2014, pp. 536-551: Springer.
 - [40] C. Choi and H. I. Christensen, "RGB-D object pose estimation in unstructured environments," *Robotics and Autonomous Systems*, vol. 75, pp. 595-613, 2016.
 - [41] K. Pauwels, L. Rubio, J. Diaz, and E. Ros, "Real-time model-based rigid object pose estimation and tracking combining dense and sparse visual cues," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Portland, USA, 2013, pp. 2347-2354.
 - [42] J. Gall, A. Yao, N. Razavi, L. Van Gool, and V. Lempitsky, "Hough forests for object detection, tracking, and action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 11, pp. 2188-2202, 2011.
 - [43] J. Shotton *et al.*, "Efficient human pose estimation from single depth images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2821-2840, 2013.
 - [44] E. Brachmann, F. Michel, A. Krull, M. Ying Yang, and S. Gumhold, "Uncertainty-driven 6d pose estimation of objects and scenes from a single rgb image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, USA, 2016, pp. 3364-3372.
 - [45] Z. Tu and X. Bai, "Auto-context and its application to high-level vision tasks and 3d brain image segmentation," *IEEE Transactions on Pattern*

Analysis and Machine Intelligence, vol. 32, no. 10, pp. 1744-1757, 2010.

- [46] F. Michel *et al.*, "Global hypothesis generation for 6D object pose estimation," *arXiv preprint arXiv:1612.02287*, 2016.
- [47] P. Wohlhart and V. Lepetit, "Learning descriptors for object recognition and 3d pose estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3109-3118.
- [48] W. Kehl, F. Manhardt, F. Tombari, S. Ilic, and N. Navab, "SSD-6D: Making RGB-Based 3D Detection and 6D Pose Estimation Great Again," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, USA, 2017, pp. 1521-1529.
- [49] V. Balntas, A. Doumanoglou, C. Sahin, J. Sock, R. Kouskouridas, and T.-K. Kim, "Pose Guided RGBD Feature Learning for 3D Object Pose Estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, USA, 2017, pp. 3856-3864.
- [50] W. Kehl, F. Milletari, F. Tombari, S. Ilic, and N. Navab, "Deep learning of local rgb-d patches for 3d object detection and 6d pose estimation," in *European Conference on Computer Vision*, Amsterdam, The Netherlands, 2016, pp. 205-220: Springer.
- [51] A. Krull, E. Brachmann, S. Nowozin, F. Michel, J. Shotton, and C. Rother, "PoseAgent: Budget-Constrained 6D Object Pose Estimation via Reinforcement Learning," *arXiv preprint arXiv:1612.03779*, 2016.
- [52] A. Krull, E. Brachmann, F. Michel, M. Ying Yang, S. Gumhold, and C. Rother, "Learning analysis-by-synthesis for 6d pose estimation in rgb-d images," in *Proceedings of the IEEE International Conference on Computer Vision*, Boston, USA, 2015, pp. 954-962.
- [53] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5-32, 2001.
- [54] G. Fanelli, J. Gall, and L. Van Gool, "Real time head pose estimation with random regression forests," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, Colorado Springs, USA, 2011, pp. 617-624: IEEE.
- [55] A. Doumanoglou, R. Kouskouridas, S. Malassiotis, and T.-K. Kim, "Recovering 6D object pose and predicting next-best-view in the crowd," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3583-3592.
- [56] A. Tejani, R. Kouskouridas, A. Doumanoglou, D. Tang, and T.-K. Kim, "Latent-Class Hough Forests for 6 DoF Object Pose Estimation," *IEEE*

transactions on pattern analysis and machine intelligence, vol. 40, no. 1, pp. 119-132, 2018.

- [57] A. Torralba, K. P. Murphy, and W. T. Freeman, "Contextual models for object detection using boosted random fields," in *Advances in neural information processing systems*, Vancouver, Canada, 2005, pp. 1401-1408.
- [58] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation," in *European conference on computer vision*, Graz, Austria, 2006, pp. 1-15: Springer.
- [59] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie, "Objects in context," in *Computer vision, 2007. ICCV 2007. IEEE 11th international conference on*, Rio de Janeiro, Brazil, 2007, pp. 1-8: IEEE.
- [60] P. Kotschieder, S. R. Bulò, A. Criminisi, P. Kohli, M. Pelillo, and H. Bischof, "Context-sensitive decision forests for object detection," in *Advances in neural information processing systems*, 2012, pp. 431-439.
- [61] A. Montillo, J. Shotton, J. Winn, J. E. Iglesias, D. Metaxas, and A. Criminisi, "Entangled decision forests and their application for semantic segmentation of CT images," in *Biennial International Conference on Information Processing in Medical Imaging*, Irsee, Germany, 2011, pp. 184-196: Springer.
- [62] P. Kotschieder, S. R. Bulò, M. Pelillo, and H. Bischof, "Structured labels in random forests for semantic labelling and object detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 10, pp. 2104-2116, 2014.
- [63] L. Wolf and S. Bileschi, "A critical view of context," *International Journal of Computer Vision*, vol. 69, no. 2, pp. 251-261, 2006.
- [64] M. Fink and P. Perona, "Mutual boosting for contextual inference," in *Advances in neural information processing systems*, Vancouver, Canada, 2004, pp. 1515-1522.
- [65] S. Avidan, "Spatialboost: Adding spatial reasoning to adaboost," in *European Conference on Computer Vision*, Graz, Austria, 2006, pp. 386-396: Springer.
- [66] M. Dantone, J. Gall, C. Leistner, and L. Van Gool, "Human pose estimation using body parts dependent joint regressors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Portland, USA, 2013, pp. 3041-3048.

- [67] V. Ramakrishna, D. Munoz, M. Hebert, J. A. Bagnell, and Y. Sheikh, "Pose machines: Articulated pose estimation via inference machines," in *European Conference on Computer Vision*, Zurich, Switzerland, 2014, pp. 33-47: Springer.
- [68] A. Bicchi and V. Kumar, "Robotic grasping and contact: A review," in *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, San Francisco, USA, 2000, vol. 1, pp. 348-353: IEEE.
- [69] J. Bohg, A. Morales, T. Asfour, and D. Kragic, "Data-driven grasp synthesis—a survey," *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 289-309, 2014.
- [70] C. Choi and H. I. Christensen, "Robust 3D visual tracking using particle filtering on the special Euclidean group: A combined approach of keypoint and edge features," *The International Journal of Robotics Research*, vol. 31, no. 4, pp. 498-519, 2012.
- [71] A. Collet, D. Berenson, S. S. Srinivasa, and D. Ferguson, "Object recognition and full pose registration from a single image for robotic manipulation," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, Kobe, Japan, 2009, pp. 48-55: IEEE.
- [72] C. F. Olson and D. P. Huttenlocher, "Automatic target recognition by matching oriented edge pixels," *IEEE Transactions on image processing*, vol. 6, no. 1, pp. 103-113, 1997.
- [73] A. Agrawal, Y. Sun, J. Barnwell, and R. Raskar, "Vision-guided robot system for picking objects by casting shadows," *The International Journal of Robotics Research*, vol. 29, no. 2-3, pp. 155-173, 2010.
- [74] M.-Y. Liu, O. Tuzel, A. Veeraraghavan, R. Chellappa, A. Agrawal, and H. Okuda, "Pose estimation in heavy clutter using a multi-flash camera," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, Anchorage, USA, 2010, pp. 2028-2035: IEEE.
- [75] H. Dong, G. Sun, W.-C. Pang, E. Asadi, D. K. Prasad, and I.-M. Chen, "Fast Ellipse Detection via Gradient Information for Robotic Manipulation of Cylindrical Objects," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 2754-2761, 2018.
- [76] H. D. D. K. Prasad, Q. Yuan, J. Zhou, E. Asadi, and I.-M. Chen, "Efficient Pose Estimation from Single RGB-D Image via Hough Forest with Auto-Context," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 7201-7206: IEEE.
- [77] H. Dong, E. Asadi, G. Sun, D. K. Prasad, and I.-M. Chen, "Real-Time Robotic Manipulation of Cylindrical Objects in Dynamic Scenarios

- Through Elliptic Shape Primitives," *IEEE Transactions on Robotics*, 2018.
- [78] G. Pavlakos, X. Zhou, A. Chan, K. G. Derpanis, and K. Daniilidis, "6-dof object pose from semantic keypoints," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, Singapore, 2017, pp. 2011-2018: IEEE.
 - [79] D. Rao, Q. V. Le, T. Phoka, M. Quigley, A. Sudsang, and A. Y. Ng, "Grasping novel objects with depth segmentation," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, Taipei, Taiwan, 2010, pp. 2578-2585: IEEE.
 - [80] L. Bo, X. Ren, and D. Fox, "Learning hierarchical sparse features for RGB-(D) object recognition," *The International Journal of Robotics Research*, vol. 33, no. 4, pp. 581-599, 2014.
 - [81] M. Schwarz, A. Milan, A. S. Periyasamy, and S. Behnke, "RGB-D object detection and semantic segmentation for autonomous manipulation in clutter," *The International Journal of Robotics Research*, pp. 612-628, 2016.
 - [82] L. Pinto and A. Gupta, "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, Stockholm, Sweden, 2016, pp. 3406-3413: IEEE.
 - [83] J. Redmon and A. Angelova, "Real-time grasp detection using convolutional neural networks," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, Seattle, USA, 2015, pp. 1316-1322: IEEE.
 - [84] E. Johns, S. Leutenegger, and A. J. Davison, "Deep learning a grasp function for grasping under gripper pose uncertainty," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, Daejeon, Korea, 2016, pp. 4461-4468: IEEE.
 - [85] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705-724, 2015.
 - [86] S. Kumra and C. Kanan, "Robotic grasp detection using deep convolutional neural networks," *arXiv preprint arXiv:1611.08036*, 2016.
 - [87] J. Glover, D. Rus, and N. Roy, "Probabilistic models of object geometry for grasp planning," *Proceedings of Robotics: Science and Systems IV, Zurich, Switzerland*, pp. 278-285, 2008.

- [88] A. Saxena, J. Driemeyer, and A. Y. Ng, "Robotic grasping of novel objects using vision," *The International Journal of Robotics Research*, vol. 27, no. 2, pp. 157-173, 2008.
- [89] Y. Jiang, S. Moseson, and A. Saxena, "Efficient grasping from rgbd images: Learning using a new rectangle representation," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, Shanghai, China, 2011, pp. 3304-3311: IEEE.
- [90] 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," *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 421-436, 2018.
- [91] D. Kalashnikov *et al.*, "QT-Opt: Scalable Deep Reinforcement Learning for Vision-Based Robotic Manipulation," *arXiv preprint arXiv:1806.10293*, 2018.
- [92] J. Mahler *et al.*, "Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics," *arXiv preprint arXiv:1703.09312*, 2017.
- [93] S. Hirose and Y. Umetani, "The development of soft gripper for the versatile robot hand," *Mechanism and machine theory*, vol. 13, no. 3, pp. 351-359, 1978.
- [94] M. Zecca *et al.*, "Experimental analysis of the proprioceptive and exteroceptive sensors of an underactuated prosthetic hand," in *Proceeding of the ICORR 2003 (The Eighth International Conference on Rehabilitation Robotics)*, Daejeon, Korea, 2003.
- [95] A. M. Dollar and R. D. Howe, "The highly adaptive SDM hand: Design and performance evaluation," *The international journal of robotics research*, vol. 29, no. 5, pp. 585-597, 2010.
- [96] M. Ciocarlie *et al.*, "The Velo gripper: A versatile single-actuator design for enveloping, parallel and fingertip grasps," *The International Journal of Robotics Research*, vol. 33, no. 5, pp. 753-767, 2014.
- [97] M. Ciocarlie and P. Allen, "A constrained optimization framework for compliant underactuated grasping," *Mech. Sciences*, vol. 2, no. 1, pp. 17-26, 2011.
- [98] T. Nishimura, K. Mizushima, Y. Suzuki, T. Tsuji, and T. Watanabe, "Variable-grasping-mode underactuated soft gripper with environmental contact-based operation," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1164-1171, 2017.

- [99] N. Rojas, R. R. Ma, and A. M. Dollar, "The GR2 gripper: an underactuated hand for open-loop in-hand planar manipulation," *IEEE Transactions on Robotics*, vol. 32, no. 3, pp. 763-770, 2016.
- [100] S. B. Backus and A. M. Dollar, "An adaptive three-fingered prismatic gripper with passive rotational joints," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 668-675, 2016.
- [101] G. Palli *et al.*, "The DEXMART hand: Mechatronic design and experimental evaluation of synergy-based control for human-like grasping," *The International Journal of Robotics Research*, vol. 33, no. 5, pp. 799-824, 2014.
- [102] M. Grebenstein *et al.*, "The hand of the DLR hand arm system: Designed for interaction," *International Journal of Robotics Research*, vol. 31, no. 13, pp. 1531-1555, 2012.
- [103] R. Ozawa, H. Kobayashi, and K. Hashirii, "Analysis, classification, and design of tendon-driven mechanisms," *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 396-410, 2014.
- [104] J. Krahn, F. Fabbro, and C. Menon, "A Soft-Touch Gripper for Grasping Delicate Objects," *IEEE/ASME Transactions on Mechatronics*, 2017.
- [105] L.-W. Tsai and J.-J. Lee, "Kinematic analysis of tendon-driven robotic mechanisms using graph theory," *Journal of mechanisms, transmissions, and automation in design*, vol. 111, no. 1, pp. 59-65, 1989.
- [106] J.-J. Lee and L.-W. Tsai, "The structural synthesis of tendon-driven manipulators having a pseudotriangular structure matrix," *The International Journal of Robotics Research*, vol. 10, no. 3, pp. 255-262, 1991.
- [107] Y.-J. Ou and L.-W. Tsai, "Kinematic synthesis of tendon-driven manipulators with isotropic transmission characteristics," *Journal of mechanical design*, vol. 115, no. 4, pp. 884-891, 1993.
- [108] H. Dong, E. Asadi, C. Qiu, J. Dai, and I.-M. Chen, "Geometric design optimization of an under-actuated tendon-driven robotic gripper," *Robotics and Computer-Integrated Manufacturing*, vol. 50, pp. 80-89, 2018.
- [109] Y. Qin, K. Zhang, J. Li, and J. S. Dai, "Modelling and analysis of a rigid-compliant parallel mechanism," *Robotics and Computer-Integrated Manufacturing*, vol. 29, no. 4, pp. 33-40, 2013.
- [110] R. Datta, S. Pradhan, and B. Bhattacharya, "Analysis and design optimization of a robotic gripper using multiobjective genetic

- algorithm," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 1, pp. 16-26, 2016.
- [111] M. Ciocarlie, F. M. Hicks, and S. Stanford, "Kinetic and dimensional optimization for a tendon-driven gripper," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, Karlsruhe, Germany, 2013, pp. 2751-2758: IEEE.
 - [112] R. V. Rao and G. Waghmare, "Design optimization of robot grippers using teaching-learning-based optimization algorithm," *Advanced Robotics*, vol. 29, no. 6, pp. 431-447, 2015.
 - [113] A. M. Dollar and R. D. Howe, "Joint coupling design of underactuated hands for unstructured environments," *The International Journal of Robotics Research*, vol. 30, no. 9, pp. 1157-1169, 2011.
 - [114] C. Gosselin, F. Pelletier, and T. Laliberte, "An anthropomorphic underactuated robotic hand with 15 dofs and a single actuator," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, Pasadena, USA, 2008, pp. 749-754: IEEE.
 - [115] T. Treratanakulwong, H. Kaminaga, and Y. Nakamura, "Low-friction tendon-driven robot hand with carpal tunnel mechanism in the palm by optimal 3D allocation of pulleys," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, China, 2014, pp. 6739-6744: IEEE.
 - [116] J. M. Inouye, J. J. Kutch, and F. J. Valero-Cuevas, "A novel synthesis of computational approaches enables optimization of grasp quality of tendon-driven hands," *IEEE Transactions on Robotics*, vol. 28, no. 4, pp. 958-966, 2012.
 - [117] A. Soetedjo and K. Yamada, "Fast and robust traffic sign detection," in *2005 IEEE International Conference on Systems, Man and Cybernetics*, Waikoloa, Hawaii, USA, 2005, vol. 2, pp. 1341-1346: IEEE.
 - [118] D. S. Barwick, "Very fast best-fit circular and elliptical boundaries by chord data," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 6, pp. 1147-1152, 2009.
 - [119] M. Saska *et al.*, "System for deployment of groups of unmanned micro aerial vehicles in GPS-denied environments using onboard visual relative localization," *Autonomous Robots*, vol. 41, no. 4, pp. 919-944, 2017.
 - [120] S. S. M. Salehian, M. Khoramshahi, and A. Billard, "A dynamical system approach for softly catching a flying object: Theory and experiment," *IEEE Transactions on Robotics*, vol. 32, no. 2, pp. 462-471, 2016.

- [121] S. Kim, A. Shukla, and A. Billard, "Catching objects in flight," *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1049-1065, 2014.
- [122] J. Su, Z.-Y. Liu, H. Qiao, and C. Liu, "Pose-estimation and reorientation of pistons for robotic bin-picking," *Industrial Robot: An International Journal*, vol. 43, no. 1, pp. 22-32, 2016.
- [123] J. A. Piepmeier and H. Lipkin, "Uncalibrated eye-in-hand visual servoing," *The International Journal of Robotics Research*, vol. 22, no. 10-11, pp. 805-819, 2003.
- [124] J. Canny, "A computational approach to edge detection," *IEEE Transactions on pattern analysis and machine intelligence*, no. 6, pp. 679-698, 1986.
- [125] D. K. Prasad, M. K. H. Leung, C. Quek, and M. S. Brown, "DEB: Definite error bounded tangent estimator for digital curves," *IEEE Transactions on Image Processing*, vol. 23, no. 10, pp. 4297 - 4310, 2014.
- [126] H. Freeman and R. Shapira, "Determining the minimum-area encasing rectangle for an arbitrary closed curve," *Communications of the ACM*, vol. 18, no. 7, pp. 409-413, 1975.
- [127] A. Y.-S. Chia, S. Rahardja, D. Rajan, and M. K. Leung, "A split and merge based ellipse detector with self-correcting capability," *IEEE Transactions on Image Processing*, vol. 20, no. 7, pp. 1991-2006, 2011.
- [128] H. Dong, I.-M. Chen, and D. K. Prasad, "Robust ellipse detection via arc segmentation and classification," in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 66-70: IEEE.
- [129] D. K. Prasad, M. K. Leung, and C. Quek, "Ellifit: an unconstrained, non-iterative, least squares based geometric ellipse fitting method," *Pattern Recognition*, vol. 46, no. 5, pp. 1449-1465, 2013.
- [130] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE transactions on pattern analysis and machine intelligence*, vol. 17, no. 8, pp. 790-799, 1995.
- [131] A. S. Aguado, M. E. Montiel, and M. S. Nixon, "On using directional information for parameter space decomposition in ellipse detection," *Pattern Recognition*, vol. 29, no. 3, pp. 369-381, 1996.
- [132] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, Miami, USA 2009, pp. 248-255: IEEE.

- [133] M. Sun, G. Bradski, B.-X. Xu, and S. Savarese, "Depth-encoded hough voting for joint object detection and shape recovery," in *European Conference on Computer Vision*, Crete, Greece, 2010, pp. 658-671: Springer.
- [134] A. Yao, J. Gall, and L. Van Gool, "A hough transform-based voting framework for action recognition," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, San Francisco, USA, 2010, pp. 2061-2068: IEEE.
- [135] R. Girshick, J. Shotton, P. Kohli, A. Criminisi, and A. Fitzgibbon, "Efficient regression of general-activity human poses from depth images," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, Barcelona, Spain, 2011, pp. 415-422: IEEE.
- [136] J. S. Vitter, "Random sampling with a reservoir," *ACM Transactions on Mathematical Software (TOMS)*, vol. 11, no. 1, pp. 37-57, 1985.
- [137] A. Criminisi, J. Shotton, D. Robertson, and E. Konukoglu, "Regression forests for efficient anatomy detection and localization in CT studies," in *International MICCAI Workshop on Medical Computer Vision*, Athens, Greece, 2010, pp. 106-117: Springer.
- [138] O. Kallenberg, *Foundations of modern probability*. Springer Science & Business Media, 2006.
- [139] P. Kotschieder, P. Kohli, J. Shotton, and A. Criminisi, "GeoF: Geodesic forests for learning coupled predictors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Portland, USA, 2013, pp. 65-72.
- [140] G. Carlsson and F. MÅŠmoli, "Characterization, stability and convergence of hierarchical clustering methods," *Journal of machine learning research*, vol. 11, no. Apr, pp. 1425-1470, 2010.
- [141] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon, "Scene coordinate regression forests for camera relocalization in RGB-D images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Portland, USA, 2013, pp. 2930-2937.
- [142] B. Drost, M. Ulrich, N. Navab, and S. Ilic, "Model globally, match locally: Efficient and robust 3D object recognition," in *2010 IEEE computer society conference on computer vision and pattern recognition*, 2010, pp. 998-1005: Ieee.
- [143] M. Quigley *et al.*, "ROS: an open-source Robot Operating System," in *ICRA workshop on open source software*, Kobe, Japan, 2009, vol. 3, no. 3.2, p. 5: Kobe.

- [144] S. Chitta, I. Sucan, and S. Cousins, "Moveit![ROS topics]," *IEEE Robotics & Automation Magazine*, vol. 19, no. 1, pp. 18-19, 2012.
- [145] G. Bradski, "The OpenCV Library," *Dr. Dobbs's Journal: Software Tools for the Professional Programmer*, vol. 25, no. 11, pp. 120-123, 2000.
- [146] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM international conference on Multimedia*, Orlando, USA 2014, pp. 675-678: ACM.
- [147] S. C. Jacobsen, J. E. Wood, D. Knutti, and K. B. Biggers, "The UTAH/MIT dextrous hand: Work in progress," *The International Journal of Robotics Research*, vol. 3, no. 4, pp. 21-50, 1984.
- [148] H. Kobayashi, K. Hyodo, and D. Ogane, "On tendon-driven robotic mechanisms with redundant tendons," *The International Journal of Robotics Research*, vol. 17, no. 5, pp. 561-571, 1998.
- [149] L. Birglen, T. Laliberté, and C. M. Gosselin, *Underactuated robotic hands*. Springer, 2007.
- [150] H. Dong, C. Qiu, D. K. Prasad, Y. Pan, J. Dai, and I.-M. Chen, "Enabling grasp action: Generalized quality evaluation of grasp stability via contact stiffness from contact mechanics insight," *Mechanism and Machine Theory*, vol. 134, pp. 625-644, 2019.
- [151] H. Dong, E. Asadi, C. Qiu, J. Dai, and I.-M. Chen, "Grasp analysis and optimal design of robotic fingertip for two tendon-driven fingers," *Mechanism and Machine Theory*, vol. 130, pp. 447-462, 2018.
- [152] A. Hassan and M. Abomoharam, "Modeling and design optimization of a robot gripper mechanism," *Robotics and Computer-Integrated Manufacturing*, vol. 46, pp. 94-103, 2017.
- [153] J. Ölvander, M. Tarkian, and X. Feng, "Multi-objective Optimisation of a Family of Industrial Robots," in *Multi-objective Evolutionary Optimisation for Product Design and Manufacturing*: Springer, 2011, pp. 189-217.
- [154] T. Laliberté and C. M. Gosselin, "Simulation and design of underactuated mechanical hands," *Mechanism and machine theory*, vol. 33, no. 1-2, pp. 39-57, 1998.
- [155] G. A. Kragten, F. C. Van der Helm, and J. L. Herder, "A planar geometric design approach for a large grasp range in underactuated hands," *Mechanism and Machine Theory*, vol. 46, no. 8, pp. 1121-1136, 2011.

- [156] L. Birglen and C. M. Gosselin, "Kinetostatic analysis of underactuated fingers," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 211-221, 2004.
- [157] T. Laliberté and C. M. Gosselin, "Simulation and design of underactuated mechanical hands," *Mechanism and machine theory*, vol. 33, no. 1, pp. 39-57, 1998.
- [158] M. Russo *et al.*, "Design and test of a gripper prototype for horticulture products," *Robotics and Computer-Integrated Manufacturing*, vol. 44, pp. 266-275, 2017.
- [159] B. Alexander and K. Viktor, "Proportions of hand segments," *Int. J. Morphol*, vol. 28, no. 3, pp. 755-758, 2010.
- [160] I. M. Bullock, T. Feix, and A. M. Dollar, "Analyzing human fingertip usage in dexterous precision manipulation: Implications for robotic finger design," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, Chicago, USA, 2014, pp. 1622-1628: IEEE.
- [161] M. Mitchell, *An introduction to genetic algorithms*. MIT press, 1998.

List of Publications

Journals

1. H. Dong, E. Asadi, C. Qiu, J. Dai, and I-Ming Chen. "Geometric design optimization of an under-actuated tendon-driven robotic gripper." *Robotics and Computer-Integrated Manufacturing* (2017).
2. H. Dong, D. K. Prasad, and I-Ming Chen, "Accurate detection of ellipses with false detection control at video rates using a gradient analysis," *Pattern Recognition*, 2018.
3. H. Dong, E. Asadi, G. Sun, D. K. Prasad, I-Ming Chen, "Real-time Robotic Manipulation of Cylindrical Objects in Dynamic Scenarios through Elliptic Shape Primitives," *IEEE Transactions on Robotics* (2018).
4. H. Dong, E. Asadi, C. Qiu, J. Dai, and I-Ming Chen. " Grasp analysis and optimal design of robotic fingertip for two tendon-driven fingers." *Mechanism and Machine Theory* (2018).
5. H. Dong, G.Sun, W.Pang, E.Asadi, D.k. Prasad, I-Ming Chen." Fast Ellipse Detection via Gradient Information for Robotic Manipulation of Cylindrical Objects." *IEEE Robotics and Automation Letters* (2018).
6. H. Dong, C Qiu, DK Prasad, Y Pan, J Dai, IM Chen." Enabling grasp action: Generalized quality evaluation of grasp stability via contact stiffness from contact mechanics insight." *Mechanism and Machine Theory* (2019).

Conference proceedings

7. H. Dong, D. K. Prasad, Q.Yuan, J.Zhou, E.Asadi,I-Ming Chen. " Efficient Pose Estimation from Single RGB-D Image via Hough Forest with Auto-Context," *IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS)*, October,1-5, 2018, Madrid, Spain.
8. H. Dong, G.Sun, W.-C.Pang, E.Asadi, D.K.Prasad,I-Ming Chen, "Fast Ellipse Detection via Gradient Information for Robotic Manipulation of Cylindrical Objects," *IEEE/RSJ International Conference on Intelligent Robots and Systems, October(IROS-RAL)*, 1-5, 2018, Madrid, Spain.
9. H. Dong, I-Ming Chen, and D. K. Prasad, "Robust ellipse detection via arc segmentation and classification," *2017 IEEE International Conference on Image Processing (ICIP)*, 66-70, 2017, Beijing, China.