Duan, Li (2023) *Robotic perception and manipulation of garments*. PhD thesis.

Robotic Perception and Manipulation of Garments

Li Duan

Submitted in fulfilment of the requirements for the
Degree of Doctor of Philosophy

School of Computing Science
College of Science and Engineering
University of Glasgow

February 2023

Abstract

This thesis introduces an effective robotic garment flattening pipeline and robotic perception paradigms for predicting garments' geometric (shape) and physics properties.

Robotic garment manipulation is a popular and challenging task in robotic research. Due to the high dimensionality of garments, object states of garments are infinite. Also, garments deform irregularly during manipulations, which makes predicting their deformations difficult. However, robotic garment manipulation is an essential topic in robotic research. Robotic laundry and household sorting play a vital role in an ageing society, and automated manufacturing requires robots to be able to grasp different mechanical components, some of which are deformable objects. Also, robot-aided garment dressing is essential for the community with disabilities. Therefore, designing and implementing effective robotic garment manipulation pipelines are necessary but challenging.

This thesis mainly focuses on designing an effective robotic garment flattening pipeline. Therefore, this thesis is divided into two main parts: robotic perception and robotic manipulation. Below is a summary of the research in this PhD thesis:

- Robotic perception provides prior knowledge on garment attributes (geometrical (shape) and physics properties) that facilitates robotic garment flattening. Continuous perception paradigms are introduced for predicting shapes and visually perceived garments weights.
- A reality-simulation knowledge transferring paradigm for predicting the physics properties of real garments and fabrics has been proposed in this thesis.
- The second part of this thesis is robotic manipulation. This thesis suggests learning the *known configurations* of garments with prior knowledge of garments' geometric (shape) properties and selecting pre-designed manipulation strategies to flatten garments. The robotic manipulation part takes advantage of the geometric (shape) properties learned from the robotic perception part to recognise the *known configurations* of garments, demon-

strating the importance of robotic perception in robotic manipulation.

The experiment results of this thesis revealed that: 1). A robot gains confidence in prediction (shapes and visually perceived weights of unseen garments) from continuously perceiving video frames of unseen garments being grasped, where high accuracies on predictions (93% for shapes and 98.5 % for visually perceived weights) are obtained; 2). Predicting the physics properties of real garments and fabrics can be realised by learning physics similarities between simulated fabrics. The approach in this thesis outperforms SOTA (34 % improvement on real fabrics and 68.1 % improvement for real garments); 3). Compared with state-of-the-art robotic garment flattening, this thesis enables the flattening of garments of various shapes (five shapes) and fast and effective manipulations. Therefore, this thesis advanced SOTA of robotic perception and manipulation (flattening) of garments.

Acknowledgements

I would like to thank Dr Gerardo Aragon-Camarasa for his supervision over the last three years, which promoted my research, presentation and writing skills. During my PhD, Dr Gerardo encouraged me to prepare talks, seminars and technical reports for the computer vision and autonomous system group (CVAS) at the School of Computing Science, University of Glasgow. I have improved my research, presenting and writing skills after these opportunities of engaging with CVAS. I would like to thank CVAS members, especially Ali, Ozan, Nicolas, Lewis and Florent. They gave me valuable tips on preparing my annual progress reports, research, paper revisions and group presentations. I could not reach this point without their help. They also helped me organise group meetings, cafe time and other group activities. They make me feel that CVAS is like a home rather than solely a research group.

I also thank my annual progress examiners, Dr Craig MacDonald and Dr Ke Yuan. They reviewed my annual progress reports and provided me with valuable suggestions on research planning and PhD progress. Their advice has boosted my research throughout my four-year PhD training. And I thank Dr Paul Sibert and Dr Paul Henderson for their suggestions in group meetings and presentations. They provided different reviews on my research, which helped me identify the problems and find solutions.

The enormous thank is given to my parents, who support me mentally and financially. Without their love, I could not even imagine today's achievements.

God bless me and guide me in my PhD and future careers. Amen!

“I am the way, the truth, and the life. No one comes to the father except through me.”

- John 14:6

"Via, Veritas, Vita", Vivat, Universitas Glasguensis!

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This section introduces this PhD thesis. Section 1.1 describes the aims and objectives of this thesis. Section 1.2 summarises this thesis scientific motivations, and Section 1.3 discusses challenges in the SOTA. Section 1.4.1 states the hypotheses and key issues with potential solutions addressed in this thesis. Section 1.5 briefly introduces the proposed system, and Section 1.6 discusses the impact of this thesis. Section 1.7 introduces the structure of this thesis.

1.1 Aims and Objectives

This thesis focuses on one of the daily-used deformable objects: garments, and investigates their geometric and physics properties and an effective garment flattening pipeline. Due to the high dimensionality of garments, traditional pick-and-place approaches for manipulating rigid objects can not fulfil the requirements for stable and successful manipulation of garments and other deformable objects. Previous approaches for manipulating garments and other deformable objects primarily include three steps: monitor object states (configurations) of garments, find manipulation points on garments, manipulate garments from these manipulation points, and finish manipulation until an expected object state (a final configuration) is achieved. Figure 1.1 depicts such process.

For effective manipulation, understanding garments' geometric and physics properties are needed because these properties determine the changes in object states (configurations) during manipulation. For example, the shapes of garments determine the positions of potential grasping points, the weights of garments determine torque forces applied by robots, and the stiffness parameters

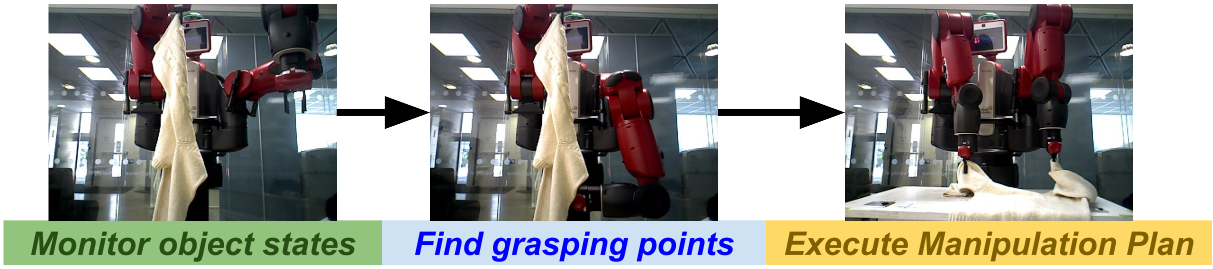


Figure 1.1: *Traditional Robotic Garment Manipulation Pipeline*: Traditional Robotic Garment Manipulation pipeline involves monitoring(recognising) garment configurations, finding manipulation points, and manipulating garments from the manipulating points. This PhD thesis investigates an effective robotic garment flattening pipeline by learning garment properties (shapes, visually perceived weights, bending stiffness parameters and area weights) to recognise their hanging configurations (known configurations).

determine the deformation patterns of manipulated garments. Prior knowledge of these properties can thus allow a robot to manipulate garments.

Previous research has developed alternatives for studying geometric properties of garments such as shapes [2, 3, 4], landmarks [4] and wrinkles [2], physics properties of garments such as bending stiffness [1][5], area weights [1], and materials [5]. Also, previous research has focused on different approaches to garment manipulation: finding grasping points by studying their hanging configurations [6, 7], analysing their wrinkles [4], or studying and predicting their configurations [8]. Instead of directly studying real garments, some researchers opted to study simulated garments to decrease time, and material costs, where reinforcement learning approaches are widely researched for robots to learn manipulation skills [9].

Manipulation strategies in previous research focus on real-time manipulation approaches, where robots update their manipulation strategies after they observe a new object state (configuration) of garments (i.e. sense-plan-act loop [10]). These updates are timely and computationally costly because updates are derived from changes in object states. Also, previous research on garment geometric/physics properties studies usually focuses on static images of garments, which cannot provide descriptions of the dynamic characteristics of garments. Therefore, this thesis aim is to investigate:

- The geometric properties (shapes) of garments using a continuous-perception paradigm (Chapters 3 and 4);
- Offline and pre-designed manipulation strategies that do not update during manipulations (Chapter 6);
- An effective robotic garment flattening pipeline (Chapter 7); and,

- The physics property of real garments and fabrics by studying the physics similarities between simulated fabrics (Chapter 5).

This thesis departs from the traditional sense-plan-act loop for manipulating garments to provide an effective garment flattening pipeline that recognises the *known configurations* of garments and selects off-line pre-designed manipulation strategies to flatten them. The *known configurations* of garments are configurations when robots hang them in the air.

1.2 Scientific Motivations

The first focus of this thesis is on transferring knowledge between simulated and real environments to learn the physics properties of garments. In real environments, measurements of physical properties such as bending and stretching stiffness parameters are difficult. Specific measurement equipment is needed, which is not easily accessible [11]. Therefore, this thesis proposes learning physics similarities between simulated fabrics to predict the physics properties of real garments and fabrics from depth images. Depth images are robust to changes between simulated and real environments. Thus knowledge learned from simulated environments is applicable in real environments. Based on this idea, this thesis demonstrates matching simulated fabrics with real fabrics and garments to predict their physics properties by learning physics similarities between simulated fabrics. Compared with SOTA, this thesis proposes that predicting physics properties of complex deformable objects (such as garments) can be achieved by learning physics similarities between deformable objects with simple structures such as fabrics.

The second focus of this thesis is on continuous perception paradigms. Previous approaches focus on detecting landmarks of garments, which include specific landmarks such as pockets, sleeves, cuffs, and collars [3] and generic landmarks such as wrinkles and folds [4]. However, this thesis departs from the idea of learning and detecting landmarks of garments to investigate learning deformations of garments being grasped from depth images. Garments of different geometric (shapes) and physics (bending stiffness) properties have different patterns of deformation when being grasped by robots, which are more straightforward and distinctive than landmark features of garments. These deformation patterns are learned by continuously perceiving garments being grasped. As a result, robots increase their confidence to predict garments' shape and visually perceived weight over time.

This thesis's third focus is investigating the *known configurations* of garments. *Known configurations* are the object states of garments when robots hang them in the air. The *known configura-*

tions only depend on grasping points due to gravity, indicating that the garments with complex configurations can be converted into simple configurations (i.e. *known configurations*). The idea of studying hanging configurations is well-researched in the literature. However, this thesis uses prior knowledge of garment geometric properties (shapes) to learn *known configurations* from depth maps (ref. Chapters 6 and 7).

1.3 Challenges

Garments are deformable objects with almost infinite object states (configurations), which deform irregularly and instantly after an external force is applied. Unlike rigid objects, such as boxes, cups and toys, which have finite object states, garments and other deformable objects usually have near-infinite and unpredictable object states. Therefore, manipulating garments requires more time to monitor their object states and derive manipulation strategies from these observations. Moreover, manipulating garments to target configurations takes more actions as the friction between operating tables and garments leads to action failures. Garments are manufactured with different textures and colours and woven with various components such as sleeves, collars and cuffs, making them structurally complex. These attributes imply that garments are difficult to be simulated in simulation engines, such as Blender [12], Unity3D [13] and the finite element method (FEM) [14] approaches, to investigate their geometric and physics properties.

Robots typically need the following steps to manipulate garments: locate grasping points, monitor garment deformations, update manipulation strategies and decide whether garments reached target configurations. Therefore, computationally expensive manipulation strategies are needed for garment manipulation, which causes inefficiency in SOTA approaches [8]. Also, garments are soft; thus, no excessive force should be applied. Therefore, robotic garment manipulation should also consider manipulation safety measures ([15]).

In summary, the following challenges exist within robotic perception and manipulation of garments:

- Due to textures, colours and components of garments, learning and predicting the geometric (shapes) and physics properties of garments from simulations is difficult because simulating them in simulation engines is time-consuming and computationally costly;
- Garments deform during manipulations to unpredictable states and configurations. Robotic garment manipulation requires monitoring garment deformations and real-time updating

of computationally expensive manipulation strategies;

- Compared with manipulating rigid objects such as cubes and boxes, more manipulations are needed for garments. Garments (and other deformable objects) have higher-dimensional object states, meaning their object states during manipulations are variable and difficult to predict. Thus, manipulating garments (and other deformable objects) is more challenging than manipulating rigid objects.

1.4 Hypotheses

This thesis devises an effective robotic garment flattening pipeline by continuously perceiving garments to predict their shape and utilising this prior knowledge of garment shapes to recognise the *known configurations* of garments. Also, this thesis investigates predicting the physics properties of real garments and fabrics by learning physics similarities between simulated fabrics. Therefore, the hypotheses of this thesis are:

- A robot can predict the shapes and visually perceived weights of unseen garments by implementing a CNN-LSTM network to learn the deformations of grasped garments and making predictions based on moving averages across the video frames of grasping garments, where accuracy is at least 10% better than using single images for predictions.

- A robot can predict the shapes and visually perceived weights of unseen garments by implementing a Siamese network [1] to learn the geometric similarities between garments and making predictions based on continuously perceiving the video frames of grasping garments with an “early-stop” strategy, where accuracy is at least 15% better than SOTA, and the pipeline requires less than 10 seconds.

- Predicting the bending stiffness parameters and area weights of real garments and fabrics can be achieved by learning physics similarities between simulated fabrics with a Siamese network, which outperforms SOTA by at least 30%;

- A robot can pick up crumpled garments from any location, recognise their *known configurations*, and select pre-designed manipulation strategies based on recognised *known configurations*, where recognition accuracy is at least 80%;

- A robot can flatten crumpled garments by picking them up from any location, predicting garment shapes, recognising the *known configurations* of garments based on predicted

[1] A Siamese network [16] is an artificial neural network that compares similarities between input variables.

garment shapes, and selecting pre-designed manipulation strategies to flatten garments, where recognition accuracy is at least 90%, and flattening garments require less than 250 seconds with a success rate over 60%

1.4.1 Key Issues and Potential Solutions

Several key issues need to be solved to answer the above hypotheses. The following sections present these key issues.

How do a robot gain confidence at predicting the shapes and visually perceived weights while continuously perceiving video frames of garments being grasped?

A robot should accumulate knowledge (i.e. gain confidence) from perceived video frames. A potential approach to realise this continuous perception paradigm is to map observations into a garment similarity map where images from garments of different shapes or visually perceived weights are clustered in different groups. When the video frames of unseen garments are mapped onto the garment similarity map, each video frame will be labelled according to which group it falls. When most video sequence frames are labelled as the same shape or visually perceived weight category, this category will be the predicted category for the unseen garment in the video sequence.

How to predict the physics of real garments and fabrics by learning physics similarities between simulated fabrics?

A potential solution is to devise a physics similarity map that encodes simulated and real garments images. A physics similarity map is a 2D manifold where images of fabrics or garments with similar physics properties are mapped to the same cluster. In contrast, images of fabrics or garments with different physics properties are mapped to different clusters. A Bayesian optimiser is used to fine-tune the physics property parameters of simulated fabrics until physics similarity distances between real garments or fabrics and simulated fabrics are minimised on the physics similarity map.

How to utilise prior knowledge of geometric properties (shapes) of garments to recognise their *known configurations* for a flattening robotic task?

The geometric properties of garments are key to determining the known configurations of garments. The geometric properties (shapes) of garments facilitate recognising the *known configurations* of garments. Prior knowledge of garment shapes can be encoded into a conditional network to recognise the *known configurations* of garments to select pre-designed manipulation strategies for a flattening task.

1.5 A Brief Overview of the Proposed Approaches

Figure 1.2 shows an overview of the proposed approaches. The proposed robotic garment perception-manipulation system consists of two parts. The robotic perception part includes a continuous perception paradigm for predicting shapes and visually perceived weights of unseen garments and a simulation-reality transfer knowledge mechanism to learn the physics properties of real garments and fabrics by learning physics similarities between simulated fabrics. The robotic garment manipulation part includes a robotic garment flattening pipeline. The robot firstly predicts garment shapes and then recognises the *known configurations* of garments based on the predicted garment shapes. Finally, the robot selects pre-designed manipulation strategies to flatten these garments.

In Figure 1.2, yellow boxes show the robotic perception part. These are divided into two components: predicting shapes and visually perceived weights of garments using the continuous-perception paradigm and predicting bending stiffness parameters and area weights from the simulation-reality transferring knowledge mechanism. Two approaches are proposed for the continuous perception paradigm: a CNN-LSTM network with a moving-average strategy and a garment similarity network with an “early-stop” strategy. Meanwhile, the robotic manipulation part is shown in the green box. This part is a robotic garment flattening pipeline in the context of this thesis. For this, a robot integrates prior knowledge of garment shapes to recognise the *known configurations* of garments and selects pre-designed manipulation strategies to flatten these garments.

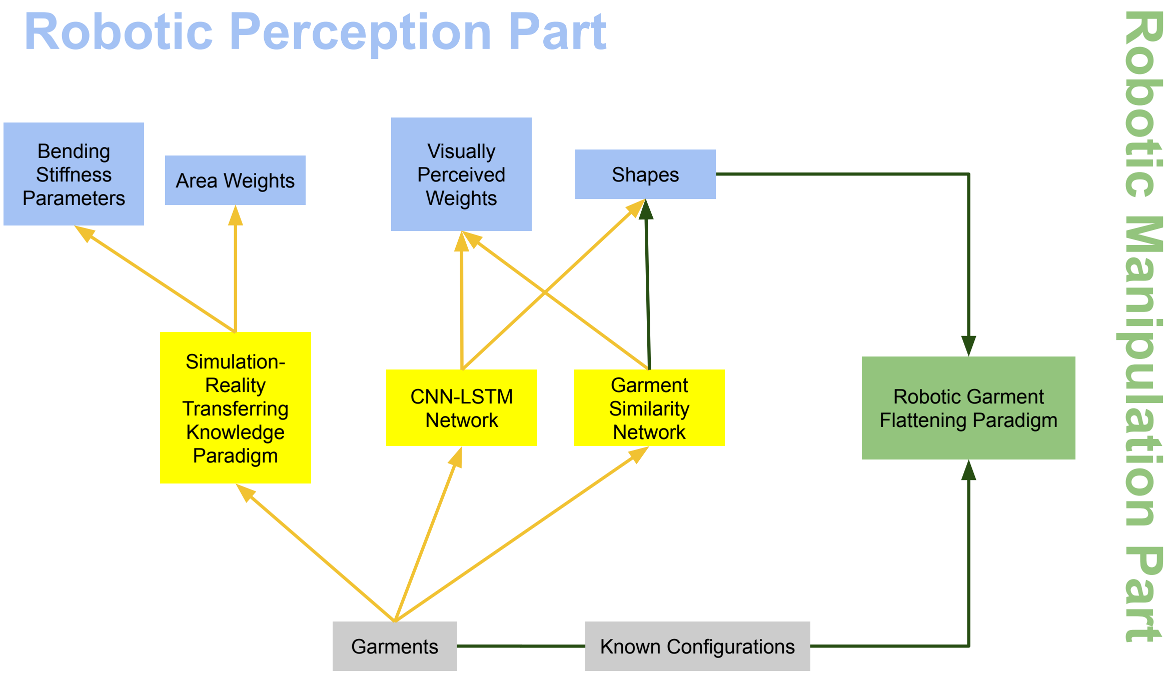

Robotic Perception Part
Bending Stiffness Parameters
Area Weights
Visually Perceived Weights
Shapes
Simulation-Reality Transferring Knowledge Paradigm
CNN-LSTM Network
Garment Similarity Network
Robotic Garment Flattening Paradigm
Garments
Known Configurations
Robotic Manipulation Part

Figure 1.2: *A Brief View of the Proposed Robotic Garment Perception-Manipulation System*

1.6 The Scientific Contributions of This Thesis

The scientific contributions of this thesis can be summarised as follows:

- A robotic perception paradigm that enables robots to continuously perceive garments to gain confidence in predicting their shapes and visually perceived weights with an "early-stop" strategy;
- A simulation-reality transfer knowledge mechanism that predicts the physics properties of real garments and fabrics by learning physics similarities between simulated fabrics;
- An effective robotic garment flattening pipeline that utilises prior knowledge of geometric properties (shapes) of garments to recognise *known configurations* of garments and select pre-designed manipulation strategies for flattening garments;

1.6.1 Robotic Demonstrations

Video demonstrations that demonstrate the above scientific contributions are as follow:

- Physics similarity network (PhySNet) Demonstration: `https://www.youtube.com/watch?v=sLdOvZjXL-A&t=53s`;

- Robotic continuous perception for predicting the shapes and visually perceived weights of garments: `https://www.youtube.com/watch?v=BJl50A1xN08`;

- Robotic garment flattening by recognising the *known configurations* of garments: `https://www.youtube.com/watch?v=jg6YbeLHrPE`;

- Robotic garment flattening by prior knowledge of garment shapes to recognise the *known configurations* of garments: `https://www.youtube.com/watch?v=j7yEbJcAgDM`.

1.6.2 List of Publications

Accepted or Published Papers

- Li Duan, Gerardo Aragon-Camarasa, "Continuous Perception for Classifying Shapes and Weights of Garments for Robotic Vision Applications", In Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 4: VISAPP, ISBN 978-989-758-555-5; ISSN 2184-4321, pages 348-355, 2022;

- Li Duan, Gerardo Aragon-Camarasa, "A Continuous Robot Vision Approach for Predicting Shapes and Visually Perceived Weights of Garments", in IEEE Robotics and Automation Letters, volume 7, No 3; pages 7950-7957, July 2022 *(nominated for Best Student Paper Award at 2022 IEEE 18th International Conference on Automation Science and Engineering (CASE))*;

- Li Duan, Gerardo Aragon-Camarasa. 2022. "Recognising Known Configurations of Garments For Dual-Arm Robotic Flattening", In 2022 4th International Conference on Robotics and Computer Vision (ICRCV), 2022, pp. 340-344;

- Li Duan, Lewis Boyd, Gerardo Aragon-Camarasa, "Learning Physics Properties of Fabrics and Garments with a Physics Similarity Neural Network", in IEEE Access, vol. 10, pp. 114725-114734.

Papers Under Review

- Li Duan, Gerardo Aragon-Camarasa, "A Data-Centric Approach For Dual-Armed Robotic Garment Flattening", Under Review by Robotics and Autonomous Systems.

1.7 Thesis Structure

Chapter 2 presents an in-depth and categorised literature review of SOTA developments in garment recognition and prediction, prediction physics properties of garments and research on robotic deformable object manipulation. Chapters 3, 4 and 5 introduce the robotic perception part of this thesis, namely a CNN-LSTM network with a “moving average” strategy and a garment similarity network with an “early-stop” strategy to predict shapes and visually perceived weights of unseen garments, a simulation-reality paradigm on predicting physics properties of garments. Chapters 6 and 7 introduce research on a robotic garment flattening pipeline based on known configurations and geometric properties of garments. Chapter 8 provides suggestions for future work and concludes this thesis.

Chapter 2

Literature Review

Robotic perception and manipulation of deformable objects have been thoroughly researched recently. Robotic perception includes recognising deformable objects' shapes and physics properties, while robotic manipulation concerns how to manipulate (flatten and fold) deformable objects. This chapter [1] summarises and discusses SOTA robotic perception and manipulation of deformable objects. SOTA limitations have been identified and discussed, and how this thesis addresses these limitations has also been provided in this chapter. Section 2.1 introduces robots, cameras and software used in this thesis. Section 2.2 reviews the development of robotic perception: garment shape classification, garment physics property investigation and garment simulations. Section 2.3 reviews the development of robotic manipulation: model-based and data-driven approaches. A summary of this thesis's literature review and advancements is presented in Section 2.5

[1] *This chapter has appeared in [17] [18][19][20] [21]. Li Duan is the first author and main contributor to these papers.*

2.1 Hardware and Software

2.1.1 Robotic System

The Baxter Robot

The Baxter robot is a dual-arm research robot which has been used in this thesis for robotic perception and manipulation of garments. Figure 2.1 shows a picture of the Baxter robot. The Baxter robot consists of a robotic head, two robotic grippers, robotic cameras, two manipulators, and wheels. The grippers of the Baxter robot are controlled by three Cartesian-coordinate parameters (X, Y, Z) and four orientation-coordinate parameters (X, Y, Z, W). The robot operating system (ROS) provides these seven parameters to locate target gripper positions. MoveIt [22] was used in this thesis for motion planning. The Baxter robot must be fixed for robotic manipulation to ensure the distance between the robot and cameras remains unchanged throughout the experiments. Therefore, the movable wheels are fixed in the lab. Robotic cameras are located on the robotic head and grippers. In this thesis, external cameras have been used to capture RGBD images, thus robotic cameras were not used and remained turned off. The Baxter robot has a larger action space than single-armed robots due to the two arms. Single-arm robots must work collectively to perform the same tasks as Baxter.

The Xtion stereo cameras

Xtion cameras were used to capture RGBD images of garments and fabrics in chapters 3, 4, 5, 6 and 7.An Xtion camera contains two cameras to capture depth information of objects (garments and fabrics in this thesis). Figure 2.2 shows a picture of an Xtion camera. An Xtion camera provides images with a size of 640 × 480 and has a sensing range of 0.8-3.5 meters. Table 2.1 compares the Xtion camera with other types of stereo (depth) cameras.

Xtion cameras are used in various robotic tasks: 3D reconstruction of objects [23], depth detection, object recognition and categorisation, and robotic motion planning [24]. Recent research is increasing the focus on studying the 3D information of manipulated objects for optimising manipulation strategies and converting the 3D information to a point cloud to study the structural characteristics of objects. Variants of Xtion cameras include Xtion2 cameras and Zed cameras. These depth cameras have two lenses in parallel. Differences in positions of lenses provide different views of an object, which are used to reconstruct depth images of objects at different

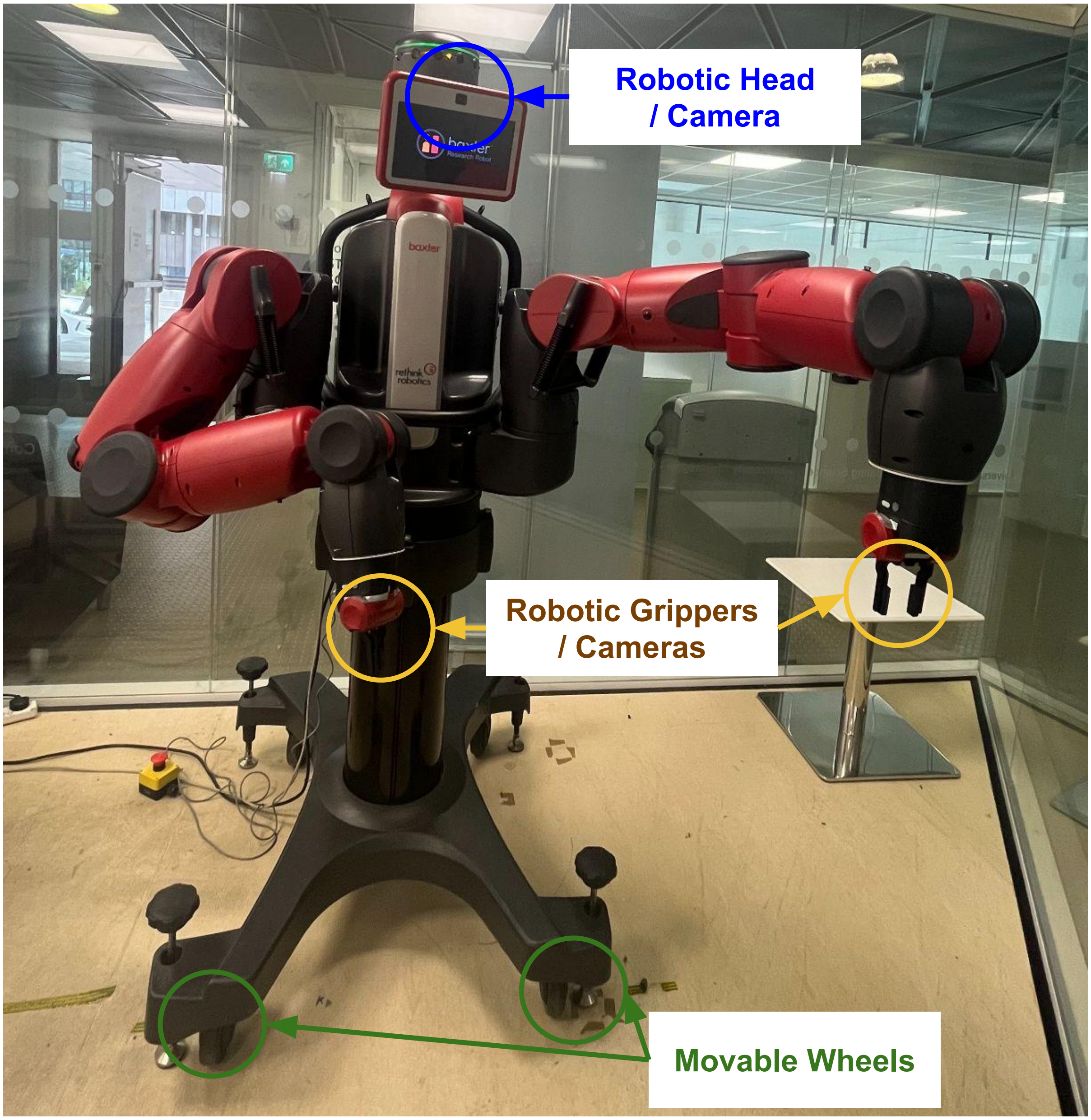


Figure 2.1: *A Baxter Robot*: The robotic grippers and cameras are coloured in orange, the robotic head and camera are coloured in black, and the movable wheels are coloured in green.

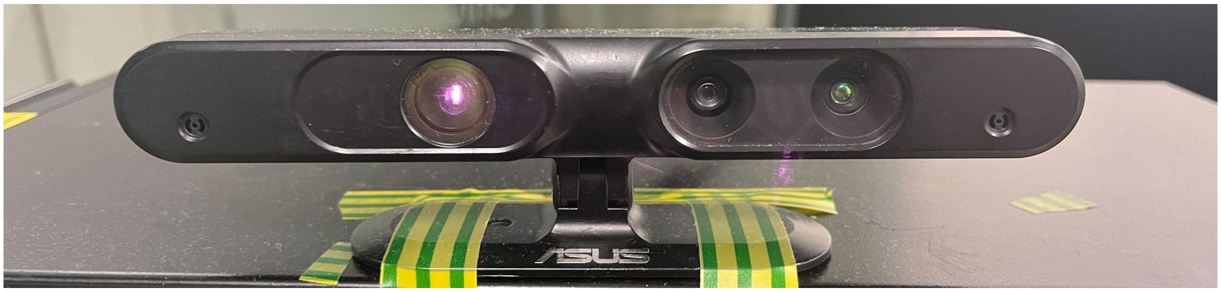


Figure 2.2: Xtion cameras have been used in this PhD thesis for capturing RGB and depth images of garments and fabrics. The Xtion cameras have a resolution of 640 × 480 and a depth range of $0.8 - 3.5$.

distances. In a depth image, the values of pixels relate to the distances between pixels and the camera.

Depth cameras can be placed on the Baxter robot itself or at a distance from the Baxter robot. In this thesis, the cameras are placed at a distance (an external point) to the front of the robot for a better view of the garments. Thus hand-eye calibration is used to align the coordinate systems of the robot and the cameras.

Name	Resolution	Depth
Xtion Camera	640×480	0.8-3.5
Xtion2 Camera	640×480	0.8-3.5
Zed 2 Camera	3840×1080	0.2-20
Intel RealSense Camera	1280×720	0.2-10

Table 2.1: *Comparison of Stereo (Depth) Cameras*

2.1.2 Operating System

The Robot Operating System (ROS [25]) is a distributed framework for controlling robots that follows a subscriber-publisher mechanism. ROS nodes are computer processes that independently execute specific commands and tasks and communicate messages with each other in the ROS system. A ROS node can publish or subscribe to other ROS nodes to receive or send messages (information) to each other, where the information is called ROS topics. ROS node-topic mechanism enables the effective exchange and processing of information between different robotic sensors (for example, cameras, touch sensors and robotic grippers). This thesis uses the ROS Kinetic version on Ubuntu 16.04 (Linux distribution).

Another feature of ROS is the launch file system. By enabling a launch file, multiple topics/nodes can be run without repetitive instantiating of each ROS node. ROS is a programexclusive system, which means a program or topic/node will be terminated if the same program or topic/node is run. ROS supports C++ and Python.

MoveIt [26] is a resource library for researchers to control robots in the ROS platform. In this thesis, controlling commands were scripted in Python and processed by MoveIt, demonstrating efficiency in the experiments. In this thesis, MoveIt was used to control robots for data collection and demonstration.

2.2 Robotic Perception

Robotic perception is the first step toward robotic deformable object manipulation. The geometry and physics properties are two essential attributes of deformable objects that influence how these objects will deform during manipulation. Geometric properties mainly include shapes, while physics properties include stiffness (bending/stretching), elasticity, and area weights. A robot can better understand the objects with prior knowledge of these attributes, which can potentially help the robotic manipulation of these objects. Therefore the robotic perception of these attributes is crucial for robotic deformable object manipulation. This section discusses the SOTA in geometric and physics properties perception in terms of their advantages and limitations. Section 2.2.1 introduces the study of the physics properties of deformable objects, and Section 2.2.2 summarises the development of technologies for predicting geometric properties of deformable objects.

2.2.1 Deformable Object Physics Properties

Physics properties of deformable objects include stiffness, elasticity, materials, damping, etc. Deforming patterns are highly related to physics properties; thus, investigating the properties of deformable objects is essential for understanding deformations, which are helpful for robotic deformable objects.

Stiffness

Stiffness is a physics property of objects that determines how stiff garments/fabrics are. Stiffness affects how garments and fabrics deform when an external force is applied. Therefore, investigating stiffness is important for robotic garment manipulations. Kawabata *et al.* [27] conceptualised human perception of the physics properties of fabrics. They categorised these physics properties as Koshi (stiffness), Numeri (smoothness) and Fukurami (fullness and softness) and also evaluated how machines quantitatively measure these physics properties. Their work provided the basic concept of fabric physics properties, enlightening the following research.

SOTA stiffness learning mainly focuses on data-driven methods ([28, 29, 5, 11, 30]), where learning stiffness is achieved from data-intensive machine learning technologies. These technologies include matching real fabrics with simulated fabrics ([28] [29]), using convolutional networks ([5] [11]), and mathematical analysis ([30]).

Miguel *et al.* [28] proposed a data-driven approach to predicting stretching, shearing and bending stiffness of fabrics by fitting the deformations of real fabrics with these of simulated fabrics with forces applied. Real fabrics are put in a specific measurement instrument where applying forces can be measured. Simulated fabrics are matched with real fabrics when applying these forces, and deformations in simulated and real environments are matched. However, this approach requires a specialised measurement instrument and only simply-structured objects (fabrics) were tested. Likewise, Tanaka *et al.* [29] also minimised the shape difference between real and simulated garments to find their stiffness. In Miguel *et al.* and Tanaka *et al.*'s approaches, if a small variation exists between simulation and reality or an unseen object is presented, they are required to simulate again as the simulation is limited to known object models.

Other researchers proposed leveraging convolutional networks for physics properties predictions besides matching simulated fabrics with real fabrics. Bouman *et al.* [11] proposed to learn the physics properties of fabrics from videos. Bouman *et al.* focused on fabric stiffness, and their approach consisted of learning statistical features of images' frequency domain of fabric videos and using a neural network to regress stiffness parameters of fabrics. Similarly, Yang *et al.* [5] proposed predicting fabrics' physics properties by learning the fabrics' dynamics from videos using a CNN-LSTM network architecture. However, these methods are constrained to fabrics with regular shapes, while the approach described in this thesis extends to garments with irregular and complex shapes.

The third approach is mathematically analysing fabric stiffness. Wang *et al.*[30] proposed parameterising the stiffness of fabrics as a piecewise linear function of the fabrics' strain tensor. That is, they sampled the strain tensor with principle strains (maximum and minimum normal strains) and strain angulars, combined as a matrix of 24 parameters for stretching stiffness (i.e. the resistance when fabrics are stretched) and 15 parameters for bending stiffness (i.e.

the resistance when fabrics are bent). To measure the stiffness of the fabrics, they opted for a FEM approach that aligned simulated meshes with the fabrics. They considered that stiffness is nonlinear, making simulations and stiffness measurements more accurate. However, the FEM method requires considerable time to compute accurately the deformation of objects which limits this approach's applicability to real-time robotic manipulation.

Although these approaches succeeded in predicting the stiffness of fabrics, which are regularly shaped (squares or rectangles), none of these approaches investigates garment stiffness, which is the main focus of this thesis. Garments are composed of irregular and complex components (sleeves, collars and pockets), which are difficult to be simulated in most SOTA simulators (e.g. Blender [12], Unity3D [13] and ArcSim [31]). Collecting a dataset with real garments is laborious and time-consuming, and measuring the stiffness of real garments is challenging since a specific instrument is required ([11]). In summary, there is a lack of research that focuses on predicting the stiffness of garments, remaining a major issue in the field and which this thesis investigates for the first time.

Materials Types and Elasticity

Deformable objects are made of different materials: rubbers, denim, cotton or nylons. The material type determines other physical properties such as stiffness, elasticity, friction and rigidness. Investigation of materials facilitates understanding these physics properties. The SOTA technologies explore materials, including data-driven approaches [32], learning from vibration [33] and force sensors [34].

Bell and Upchurch *et al.* [32] constructed a large-scale material database called Materials in Context Database (MINC). This database consists of 23 categories, showing different materials in different contexts. A convolutional neural network was proposed, including a patch-material classification and a material recognition and segmentation combined with a fully connected conditional random field (CRF, [35]). They demonstrated the performance of the proposed network in segmenting images into parts and recognising materials in the parts. However, most objects in images are rigid objects, while the network performance on deformable objects was not tested.

Instead of constructing or applying databases, specific instruments have been used by some researchers. Davis *et al.* [33] chose to investigate deformable objects' physical properties in terms of their vibration frequencies. That is, they employed a loudspeaker to generate sonic waves on fabrics to obtain modes of vibration of fabrics and analysed the characteristics of these modes of vibration to identify the fabrics' materials. The main limitation of this approach is the use of high-end sensing equipment, which would make it impractical for a robotic application. Similarly, Arriola-Rios *et al.* [34] suggested learning materials of sponges by using a force sensor mounted on a finger in a robot gripper. The finger pressed a sponge to measure the applied force, which was then used to learn the material properties and to predict the sponge's deformation.

These technologies require specific instruments (vibration or force sensors) or a large-scale database. In robotic applications, robots do not support all of these instruments. For example, the instrument to measure the stiffness parameters of fabrics in [11] can not be easily mounted on robots. Thus, some of the mentioned technologies are not practical for robotic deformable object manipulation. Technologies that do not rely on specific instruments and are easily implemented in robotic environments should be promoted in robotic deformable-object manipulation research.

Elasticity measures how well deformable objects recover to their initial states after applying external forces. Measurements of elasticity also facilitate robotic deformable object manipulation. Unlike stiffness and material types, measuring elasticity usually relates to measuring Young's modulus of objects. SOTA technologies include studying real and simulated objects and FEM-based approaches.

To study the elasticity of objects, Senguapa *et al.* [36] has proposed an approach where a robot presses the surface of objects and observes objects' shape changes in a simulated and a real environment. They aimed to find the difference between Young's modulus of the simulated and real objects to estimate the object's elasticity and estimate forces applied to the object without any force sensor.

Instead of studying the elasticity of objects by matching real-simulated objects, Yang *et al.* [37] introduced a non-invasive prostate-elasticity measurement approach. This approach used a finite-element-based bio-mechanical model generated from medical images, local displacements and an optimisation-based framework. A statistically-based and multi-class learning method was proposed to classify T-stage and Gleason scores based on patient ages and prostate elasticity. This research reveals the importance of the physics properties of tissues in cancer diagnoses.

These two approaches inspire that learning elasticity of real objects can be achieved by learning from simulated objects. Elasticity (and other physics properties) is easily accessible in simulated environments but difficult to be measured in real environments. However, the problem with learning from simulated environments is that complex structured objects (such as garments) can be challenging to be simulated or simulating these objects require a large amount of time. The balance between easily accessible elasticity (and other physics property parameters) data and simulation difficulties is still an unresolved problem in SOTA.

Other Physics Properties

Other physics properties include reflectance ([38]), rigidness ([39, 40]), damping ([41]) and roughness ([42]). These physics properties also relate to object deformation when robotic manipulation is conducted. In these technologies, predicting the physics properties of real objects from simulated objects is implemented in [41, 1], demonstrating efficiency in physics property predictions. Bhat *et al.* [41] proposed an approach to learning the physics properties of clothes from videos by minimising a squared distance error (SSD) between the angle maps of folds and silhouettes of the simulated clothes and the real clothes. However, their approach observes high variability while predicting physics properties of clothes such as shear damping, bend damping and linear drag. Likewise, learning from simulated objects to predict the physics properties of real objects has been proposed by Runia *et al.* [1]. They learnt physics similarity distances between simulated fabrics. That is, they predicted the physics properties of real fabrics, where they decreased physics similarity distances between real and simulated fabrics by fine-tuning parameters of simulated fabrics via a Bayesian optimiser. Their approach paved the way for a novel alternative that frees a network from simulation-reality approximations such as [30] and extends to regular shape fabrics, of which deformations are more complex, e.g. [43] [34], and [44].

These two technologies demonstrate that bridging gaps between simulated and real objects is an effective way to investigate the physics properties of deformable objects. However, these technologies that investigate the elasticity of deformable objects are either investigating only simple objects (e.g. fabrics and sponges), or they are not precise [41]. Simulating garments in simulation engines is computationally costly and time-consuming; it is not practical to simulate garments directly in these engines. For SOTA, leveraging simple-structured and simulated objects to learn the physics properties of real and complex objects is still lacking.

2.2.2 Deformable Object Geometric Property (Shape) Studies

This section introduces previous research on garments' geometric properties (shapes). Garment object states (configurations) during robotic manipulation are highly related to garment shapes, thus predicting garment shapes contribute to successful robotic garment manipulations. Previous research mainly used three main strategies: surface features, grasp and re-grasp (interactive perception) and convolutional neural networks. This section summarises these approaches and discusses their limitations.

Interactive Perception

Garment shape understanding and recognition is a challenge in robotic research. Garments consist of various components - collars, sleeves, pockets and buttons, and have different material textures - denim, broadcloth, seersucker, and corduroy. These attributes of garments complicate garment shape recognition. However, garment attribute understanding is essential for effective robotic garment manipulation, especially for recognising poses or configurations of garments. SOTA technologies for garment shape recognition include interactive perception ([45, 46, 47, 2, 3, 4]), surface features ([48, 49, 50, 4, 3, 2]) and convolutional neural networks ([51, 52, 53, 54, 55, 56, 57, 58, 59, 60]). This literature review discusses these technologies and their limitations.

The working assumption for interactive perception is that a robot can improve its perception (recognition) of garment shapes by interacting with them (grasping and flipping). Researchers argue that the robot can increase its accuracy in garment shape recognition with a better perception of garment interactions. Willimon *et al.* [47] proposed predicting garment shapes by comparing garments with known garments in a database and manipulating garments in an interactive perception paradigm. Silhouettes, edges and other low-level image measurements served as features for classification. Garments are re-grasped multiple times for better classification results. Their approach demonstrated that the garment-interacting approach outperforms the non-interacting approach, paving the way for future researchers on garment shape prediction. However, multiple grasps of garments are needed in experiments, which means the robot took a long time to recognise garment configurations. Likewise, Doumanoglou *et al.* [45] proposed using a random decision forest and probabilistic planning approach to recognise garment shapes and flatten garments. A robot firstly grasps garments from random grasping points and then re-grasps garments from the lowest points. Garment shapes were predicted from the second grasping using random decision forest trees trained from depth images of garments. Then grasping points for flattening garments were also found and used to flatten garments. They obtained high shape prediction accuracy from the experiments. However, the limitation in their experiments is that the robot recognised garment's shapes after several grasps, which is time-consuming.

Meanwhile, Chi *et al.*[46] proposed estimating the poses of garments by completing their shape from single images. The authors allowed a robot to grasp and drop garments to learn their poses.

However, they only captured images after the garment was grasped and hanging from the robot's gripper. Also, Sun *et al.* [2] proposed a Gaussian process classifier to predict unseen garment shapes while the robot interacted with them. That is, the robot in their experiment shakes or flips and then drops garments on a table to obtain a new state to increase the classification score. If the classification score of a garment is above a threshold, the garment is sorted based on its shape. This approach, therefore, demonstrated that interacting with garments enables an autonomous system to improve its prediction confidence over interactions and leads to higher classification accuracies.

Departing from the idea of garment recognition with single images in [47] [45] and [46], Martinez *et al.*[3] introduced the concept of continuous perception to enable a robot to predict shapes by continuously observing video frames from an Xtion depth-sensing camera rather than single image frames. They showed higher accuracy in predicting unseen garment shapes compared to [2] and [4]. However, the limitation in [3] is that they let the robot observe the entire video sequence before a decision can be made, which means that the robot takes a significant amount of time to predict a garment shape category, and this is given by the length of the video. In their work, they sample a garment for approximately 6 seconds which consists of sampling the garment from a crumpled to a hanging state.

These interactive-perception technologies demonstrate that a robot can improve its perception of garment shapes by interacting with robots. Martinez *et al.* [3] advanced the interactive perception technologies by introducing a continuous-perception mechanism. However, applying interactive perception is time-consuming, considering several grasps/flips/rotates are needed for such a mechanism. "Speeding up" the interactive perception is needed but lacking in SOTA.

Surface Features and Neural Networks

Another research direction is investigating the surface features of garments to understand and recognise their shapes. Garments are deformable objects, indicating that wrinkles, folds and overlaps are easily formed. Depending on different garment shapes and other attributes, these surface features are formed differently. Studying these surface features can facilitate a robot to recognise garment shapes. Willimon *et al.* [49] proposed learning a "Low-Level-Characteristics-Selection Mask-High Level" (L-C-S-H) network to predict garment shapes. Low level means features such as 2D shapes, edges, textures, colours and 3D shapes. Characteristics mean mid-level features such as pockets, collars and hems. The Selection Mask is a vector that indicates which characteristics/features are most valuable for each shape. Their approach found local features/Characteristics (Middle Level) contribute most to classifying garments.

To advance and promote Willimon's idea, Sun *et al.*[4] presented an approach in which local and global features are extracted from single images and used to predict unseen garment shapes. This approach uses local and global visual characteristics of garments, such as wrinkle features, for shape prediction. Compared to [51], their approach does not require interactions with garments, allowing it to be faster to predict shapes and robust while presenting unseen samples. However, prediction accuracies are constrained by the inability of the robot to interact with the garments, and no new knowledge can be captured.

Some researchers chose to take advantage of simulated data to overcome the difficulties in collecting real data. Li *et al.* [50] introduced predicting real garment shapes from simulated garments. They constructed a codebook by extracting features from the depth images of simulated garments and using sparse coding and dictionary learning technology. The constructed codebook was then used to predict the shapes of real garments, of which a Kinect camera captured depth images. Depth image contains more generic garments features than RGB images, which are affected by illumination conditions. Also, using depth images can close gaps between real and simulated environments.

The surface features can be converted to a "vocabulary" for further understanding garment shapes. Ramisa *et al.* [48] proposed a "Bag of Visual Words" algorithm to combine the appearance and 3D information of garments through the following steps. Firstly, descriptors are extracted from the appearance or depth images of garments, which are then quantised by a vocabulary, pooled and learned from a histogram of visual words. This procedure can be either applied to whole images or only regions of interest on images. The approach was benchmarked with other approaches, showing effectiveness in garment classification. However, single-shot (single-image) approaches are discussed as difficult tasks in their research and are only effective for a few garment types (e.g. jeans and shirts).

Detecting and learning surface features for garment shape understanding and recognition provides a faster solution than interactive and continuous perception. However, these technologies usually have a less-expected performance than interactive and continuous perception and often fail to be valid on unseen garments. Unseen-garment shape recognition is core in robotic garment manipulation because, in most robotic applications, garments and other objects are unseen to robots. The balance between time and accuracy is still challenging in the SOTA.

Apart from learning surface features with traditional algorithms to understand and recognise garment shapes, some researchers suggested constructing neural networks for garment shape (and landmark) prediction. These technologies involve training neural networks with established or newly collected databases. The neural networks learn high-level representations from garments and succeed in unseen garment prediction (a limitation in learning surface features). SOTA train neural networks on either newly collected or large-scale established databases. Mariolis *et al.* [51] devised a hierarchical convolutional neural network to predict the categories of garments and estimate their poses with real and simulated depth images. In 2015, their work pushed the accuracy of the classification from 79.3% to 89.38% with respect to state of the art. However, the main limitations are that their dataset consists of 13 garments belonging to three categories, and they tested their network only on seen garments. Similarly, Gabas *et al.* [52] proposed a deep-learning-based garment shape prediction by using depth maps/images. Garments hung by a robot were captured by a depth camera, where depth maps/images were used to train a convolutional neural network (CNN) to predict garment shapes. However, in their experiments, a robot rotates garments providing different views to enable the CNN to predict shapes, which is time-consuming. In Gabas *et al.*'s recent work [53], instead of training a CNN on real garment images, they trained a CNN on both real and simulated garment images. Collecting real data requires time and effort, indicating that training on simulated garment images is more effective. Prediction accuracy was improved compared with that of [52]. However, this project still needs garment rotation, causing time overheads to recognise garment shapes.

Departing from the idea that only takes single-formated data to train neural networks (RGB/ Depth images), Kampouris *et al.* [54] chose to take different sensor measurements to train a neural network. They proposed a multi-sensorial approach for predicting garment attributes: shapes, materials and patterns. They captured RGB images, depth images, photometric data and tactile data, which are separately processed by a convolutional neural network. Then, extracted features for each modality are fused to classify garment attributes. Multi-sensorial fusion provides diverse information about garments for garment attribute prediction. However, they noted that information from static garment images (no interaction with garments) did not demonstrate a high prediction accuracy (82%).

The above technologies mostly proposed simple convolution neural networks to learn garment shapes, while there are technologies that train deep neural networks on large-scale databases. The most prominent case is the deep fashion network. Liu *et al.* [55] constructed a large-scale garment database called DeepFashion, which consists of garments with different attributes: shapes, textures, fabric materials, parts, and styles. The database size is 800,000 images of different garments labelled and landmark-annotated. They trained a FashionNet on DeepFashion for garment attribute prediction and landmark detection. Their approach has two main contributions: constructing a large-scale database that can be used by following researchers studying garment attributes and a deep neural network that outperformed SOTA.

Meanwhile, Ziegler *et al.* [59] proposed a data-driven approach for detecting garment fashion-landmarks and predicting garment shapes. The neural network in their approach consists of three parts: orientation branch, landmark branch and attention branch. It was trained on a large database (DeepFashion [55]) and a small data-augmented database. They found that their approach outperformed the SOTA on landmark detection and garment shape prediction of unseen garments (90%). Their work demonstrated that the augmented dataset improved the performance of the proposed network compared with a small-scaled manipulation-specific dataset. Finally, Gustavsson *et al.* [60] proposed a garment shape prediction and landmark detection paradigm. They designed and implemented a novel attention-based network consisting of a rotation invariance encoder, a landmark localisation part, an attention branch, category-aware spatial attention and channel attention. They tested their network on the CTU dataset [61] and the DeepFashion dataset [55] and outperformed the SOTA for garment shape prediction (90.48%) and landmark detection. The network was then implemented for a robot to conduct flattening tasks.

Attention-based networks are widely used in SOTA for garment shape and landmark detection. Attention-based networks are proposed based on recurrent neural networks (RNNs). RNNs learn changes in sequential information (garment images) and predict future information. SOTA on RNNs include LSTM ([62]), bidirectional recurrent neural networks (BRNNs, [63]) and attention-based networks (for example, [64] and [65]). Attention-based networks “pay attention” to parts of features in feature maps rather than whole features to increase the efficiency in garment learning. Wang and Xu *et al.* [57] proposed a deep neural network to detect fashion landmarks and predict garment attributes. The network features two “grammars”: a kinetics-like domain dependency grammar and a symmetry grammar for the bilateral symmetry of garments. Bidirectional Convolutional Recurrent Neural Networks (BCRNNs) are implemented for landmark awareness (an attention-based network) and a category-driven mechanism. The landmark awareness focuses on local features of garments, while the category-driven mechanism focuses on global features. Their approach introduced an attention mechanism in the network, improving landmark detection and garment attribute prediction. Likewise, Liu *et al.* [58] proposed another attention-based deep neural network for landmark detection, garment shape prediction and garment attribute prediction. In their network, a landmark prediction branch was used for landmark detection, which also served as a landmark-driven attention mechanism for garment shape prediction and attribute prediction. This work and [57] demonstrated the benefits of implementing an attention mechanism in deep neural networks for garment shape prediction, garment attribute prediction, and landmark detection. However, these attention-based deep neural networks require a large-scale garment database to train, which cannot be easily achieved in a robotic setting. The databases were collected from online sources such as Google and online stores in street or shopping scenarios, which differ from robotic manipulation scenarios. Moreover, these approaches did not achieve a high overall accuracy score (around 50 to 60%).

The applications of neural networks in garment shape understanding and recognition gained some success, especially networks with attention-based mechanisms that demonstrate generalisation to garments in different scenarios. However, training these deep neural networks requires large-scale databases, which are difficult to be constructed. Using established databases is not widely applicable for robotic garment manipulation, as robotic manipulation environments differ from the environments where these databases were constructed (streets, online shopping malls and the Internet). Therefore, constructing appropriate databases for robotic garment manipulation is needed in SOTA.

2.3 Robotic Manipulation

The ultimate goal of this research is to design an effective robotic garment manipulation (flattening) pipeline. Research efforts in robotic deformable object manipulation are mainly divided into model-based and data-driven approaches. Model-based approaches find models for deformable objects, and manipulation plans are derived from these models. Data-driven approaches train robots to learn manipulation skills from reinforcement learning and imitation learning. These manipulation skills are learned in simulation environments and applied in real environments (most approaches use domain randomisation to bridge the gap between simulated and real environments). In contrast, data-driven methods find grasping points on deformable objects by learning their configurations (usually hanging configurations) from their edges and vertices and folding axes. Section 2.3.1 introduces the current developments on model-based approaches, Section 2.3.2 summarises data-driven approaches to reinforcement and imitation learning, and Section 2.3.3 discusses the research of data-driven approaches on finding grasping points for deformable object manipulation.

2.3.1 Model-Based Approaches

This section introduces model-based approaches for robotic deformable-object manipulation. Researchers use different strategies to define models for deformable objects, which monitor object states and update manipulation strategies. There are several approaches for defining models: the finite element method (FEM), the mass-spring system, point cloud, topology representation, simulation etc. This section summarises these approaches and discusses their limitations.

The Finite Element Method (FEM)

Model-based manipulation approaches define models for deformable objects, which monitor and predict deformations. Manipulation strategies are derived from these defined models and used for manipulation. The finite element method (FEM) is widely used in these approaches. FEM uses simulated meshes to construct models in software engines (blender, unity3D, and PyBullet). The movements of each mesh are calculated according to forces and manipulations applied to objects, which are used for monitoring and predicting object deformation.

Lin *et al.* [66] claimed that picking up soft 3D objects (bottles) can be achieved by modelling objects with a finite element method (FEM) and lifting objects, after passing a 'liftability test'. They investigated modelling objects with FEM, but these objects are simple. Meanwhile, Cui *et al.* [67] introduced a virtual hand modelling for grasping deformable objects. Both contact forces and deformation modelling were considered for grasping deformable objects. For this, they used a non-linear contact force model and a beam-skeleton model to model contacts between the hand and deformable objects. However, this approach only simulated cylindrical objects, while garments and other complex objects are difficult to model.

Similarly, Jia *et al.* [68] proposed a model-based planar object squeezing approach and analysed whether objects would stick or slip from the robot's grippers. The objects were simulated by a finite method analysis with meshes, where an event-driven algorithm tracked and modelled contacts. Tracked contacts served as feedback for deformation updates and aimed at finding a stable squeeze strategy for planar objects. However, planar objects (2D objects) have limited object states, which can be easily modelled by the finite element method, but most objects in robotic manipulation tasks are 3D or have a high dimensionality of object states.

FEM is a direct and straightforward way to model deformable objects; however, monitoring and predicting object deformations require massive computations and large amounts of time. Simply-structured objects, such as sponges, bottles and balls, are easy to model, monitor and predict. However, complex structured objects, such as garments, are difficult with FEM as garments consist of various components (collars, sleeves and pockets). Even if garments are successfully modelled with FEM, monitoring and predicting their deformations is time-consuming and computationally costly. Farmaga *et al.* [69] evaluated the complexity of FEM by testing running time and computer memory. They find that applying FEM to an average object (a plate) requires one million elements and 500,000 nodes, requiring 10 minutes and 1.89 GB of computer memory. Meanwhile, Tarrier *et al.* [70] found that simulating a garment with FEM requires 5 minutes. However, simplified modelling, such as topological representations, only requires less than ten topological representation points to learn a rectangular deformable object (a towel) in

Strazzeri *et al.* [71]'s work.

Mass-spring System

Another model-based approach is the mass-spring system (MSS). MSS simplifies object models by using masses and springs. Compared with FEM, MSS can reduce computations but can not be as accurate in modelling objects. Nabil *et al.* [72] introduced a robotic muscle separation process that consists of both physical modelling and visual perception. A mass-spring system modelled meats, and physics parameters were set based on rheological tests. Compared with garments and fabrics, meats deform less when a force is applied; thus they are more suitable for mass-spring modelling. Mass-spring modelling provides feedback on meat cutting and updates models when a force is applied. Similarly, Zaidi *et al.* [73] proposed a non-linear mass-spring system for modelling cubes and spheres and implemented a robot to grasp these objects. The contact forces applied by the robot were calculated from the MSS models to ensure stable grasps. However, only grasping manipulations were experimented with, while other manipulations, such as squeezing and cutting, were not experimented with due to modelling limitations.

The mass-spring system provides an alternative to FEM, reducing computational complexity in simulations. However, for SOTA, like FEM, most approaches were only experimented with simple objects (e.g. cubes and spheres), meaning that modelling complex objects (e.g. garments) is still challenging with the mass-spring system.

Point Cloud

Modelling deformable objects with FEM and MSS requires massive computations and is time-consuming. Instead of simulating objects with these approaches, some researchers proposed to represent deformable objects with point clouds. Point clouds are data points generated from depth maps of objects. Each data point contains a 3D coordinate (X, Y, Z), calculated from its distance with a stereo camera that generates the point clouds. Point clouds are directly computed from depth images from a stereo camera (for example, the Xtion camera in section 2.1.1), while FEM and MSS are computed with simulated meshes. Point clouds are a direct mapping from depth images, while FEM and MSS are computed as a complicated combination of different meshes. Therefore, Point clouds can be constructed faster than FEM and MSS.

Simeonov *et al.* [74] proposed that deformable objects can be manipulated by representing objects using point cloud rather than object models and calculating motion strategies to estimate transformations between the object's initial and goal configurations. Model-free physics and deformation learning do not require learning actual object physics properties but *conceptualising* how objects can be deformed when an external force acts into the object. The above methods are, however, constrained to regular patterns of shape changes. Similarly, Schulman *et al.* [75] tracked deformable objects from a sequence of point clouds, where the tracking algorithm is based on a probabilistic generative model which takes the point clouds of object observations and physics properties of objects and environment as inputs. An expectation maximisation algorithm was proposed to update state estimation at each time step. Ropes, towels, and sponges are tested in their experiments. However, one limitation in their experiments is that the tracking cannot be recovered if an estimated state is far from the true state.

Likewise, Lin and Wang *et al.* [76] suggested learning a pixel-based dynamics model from the cloud points of deformable objects, where the model is called a visible connectivity dynamics model. They then learned the visual points connected to the underlying cloth mesh. Finally, they constructed a dynamics model over a visual connectivity graph, which infers actions for flattening garments. The pipeline has been trained in a simulated environment and succeeded in real-environment testing. However, the tested objects are only simple objects such as towels and T-shirts because the modelling of complex objects such as long-sleeved shirts or sweaters will potentially fail in experiments. Finally, Tawbe *et al.* [43] proposed simulating sponges via a neural gas fitting method [77] rather than simulating meshes. They studied and predicted the shapes of deformable objects without prior knowledge about the objects' material properties by applying the neural gas fitting on simplified 3D point-cloud models. These 3D point-cloud models focused on the parts of an object that had been deformed to improve learning. Their approach required a multi-step learning process to simplify the models and find the deformed parts. However, this approach was tested only on objects with simple geometries.

Point clouds solve the problems of FEM and MSS and demonstrate effectiveness in predicting deformations. Although point clouds require using stereo cameras, these cameras are easily accessible and can be mounted on robots. Therefore, point clouds are widely applied for robotic deformable object manipulation. However, point clouds are direct observations from cameras; thus the quality of cameras and environmental changes can impact the performance of point clouds.

Topology Representation

To further simplify deformable object modelling, some researchers leveraged topological representation. In topological representation, objects are represented merely by vertices and edges rather than meshes. In the topological representation, no two edges will touch, and no edge passes through a vertex different from its endpoints. Topological representation simplifies objects by defining “skeletons” of objects.

Moletta *et al.* [78] proposed learning topological representations of garments for robotic garment flattening and folding tasks. Only skeleton and contour graphs are used rather than RGB images. Garments, therefore, are simplified into skeletons and contours rather than full images. A diffusion algorithm augments the skeleton graphs, and a leaf diffusion algorithm augments the contour graphs. They found no significant drop from RGB visual representations to skeleton and contour representations on downstream classification tasks. The skeleton graphs are used for flattening tasks, and the contour graphs are used for folding tasks. This approach simplifies garments into simple skeletons and contours, reducing the structural complexity of garments. However, the accuracy of downstream classification is not high (48%), thus similar garment items will be misclassified, leading to robotic task failures. Meanwhile, Antonova and Varava *et al.* [79] claimed to construct topological representations of highly deformable objects and proposed an approach to track the evolution of topological states over time. Also, the topology and topological evolution of the scene can be recovered from its point clouds. Finally, they introduced a predictive model where a sequence of point cloud observations are input, and a sequence of future topological states are output conditioned on targets or future controlling actions. They found novel simplified topological representations for deformable objects and a predictive model for predicting their future topological states, advancing previous research on FEM or mass-spring models. However, their approach was only validated in a simulated environment.

Topological representations succeed in simplifying deformable objects compared with FEM, MSS and point clouds and have been used in robotic garment manipulation. However, the simplifications of garments cause a robot to find difficulty in recognising these garments because details and features of garments are removed during the simplifications. Further research on balancing between simplifications and features of garments and other deformable objects is needed.

Others

There are other simplified representations apart from FEM, MSS, point clouds and topological presentations. Miller *et al.* [80] proposed a simplified 2D polygon representation for garments, which consists of two models: skeletal and folded models. Each model is constructed by four components: a landmark generator, a contour generator, a legal input set and a transformation generator. The simplified representations are input to an algorithm which outputs the corresponding manipulation strategies to fold garments. Their approach successfully enables a robot to manipulate garments of various shapes. However, fitting 2D polygon representations with garments need a complicated process involving operations from humans. Meanwhile, Guler *et al.*[81] also aimed to learn the deformation of soft sponges, but they proposed a Mesh-less Shape Matching approach, which comprises learning linear transformations between deformed objects.

Instead of simplifying deformable-object representations, some researchers proposed using neural networks to learn models for deformable objects. Yu *et al.* [82] introduced a model-based simulation approach to classifying the outcomes of robot-assisted dressing tasks. They trained hidden Markov models (HHMs) with simulated haptic data generated from an Nvidia PhysX simulator to classify the outcomes and tested the trained models in a real environment. They achieved high accuracy in classification (92.38%) in classifying the outcomes of robot-assisted dressing tasks. But their approach required a long time to optimise the simulator's parameters (16.8 hours). Likewise, Tanaka *et al.* [83] proposed a data-driven based deep-learning approach for robotic towel manipulations. They introduced an encode-manipulation-decode network (EMD Net) that takes initial and target configurations of garments as input and outputs corresponding motion planning. A sequence of actions is output without step-to-step motion planning. However, manipulation success depends on the structural complexity of manipulated objects, and only simple manipulations are demonstrated (such as folding an already flattened towel). Also, Yan *et al.* [8] introduced a model-based towel and rope flattening based on a contrastive estimation. A contrastive forward model (CFM) is proposed, which decides whether a predicted future object state belongs to inputting object state with a specific action. In their tests, images of towels and ropes with crumpled starting configurations and goal configurations are input into the CFM, which outputs a sequence of object transformations between starting and goal configurations and corresponding actions. Their model achieved the best performance among other forward models. Still, they only validated their approach with simple objects (towels and ropes), and it took about 15 minutes to complete the task.

These approaches provide SOTA simplifying representations for deformable objects and leverage neural networks to model deformable objects. However, most approaches focus on simple deformable objects rather than garments, and some neural-network approaches require a long time to train and test in experiments. In conclusion, effective approaches for robotic garment manipulations are still lacking in the SOTA.

2.3.2 Data-Driven Approaches-Reinforcement and Imitation Learning

Reinforcement and imitation learning are two approaches to data-driven robotic deformable-object manipulation. Instead of defining models for deformable objects, robots usually learn manipulation skills from simulated environments. There are two strategies: human/robotic demonstrations-imitation learning and reinforcement learning. This section introduces technical achievements for applying imitation learning or/and reinforcement learning to robotic deformable-object manipulation and discusses their limitations.

Demonstration and Imitation Learning

Instead of defining, representing or simplifying models for deformable objects introduced in section 2.3.1, robots can learn manipulation skills for deformable objects with imitation and reinforcement learning in simulated or real environments. As discussed in section 2.3.1, defining or finding models for representing deformable objects is computationally expensive and requires a large amount of time, while applying imitation and reinforcement learning can avoid this problem. These reinforcement learning approaches are divided into demonstration and imitation learning and reinforcement learning in simulated environments.

Robots learn from human or robotic demonstrations to gain skills in manipulating deformable objects. Compared with training robots in simulated environments with reinforcement learning, skill learning converges faster with demonstrations and imitation learning. Pignat *et al.* [84] have proposed to encode sensory information and motor commands as a joint distribution in a hidden semi-Markov model, of which parameters are learned from a set of human demonstrations. Each model set represents a sensorimotor pattern whose sequencing can produce complex behaviours.

Meanwhile, Huang *et al.* [85] introduced using appearance information for finding wrapping functions in non-rigid registration for manipulating deformable objects. A non-registration is computed between the starting scene in a human demonstration and the scene at testing time, which is extrapolated to find a wrapping function to “wrap” the demonstration trajectory to generate a proposed trajectory (in the testing scene) to manipulate deformable objects. Their approach features appearance information from deep learning to improve the quality of non-rigid registration, which better registers areas of interest on deformable objects, such as towel corners, towel edges and rope crossings. However, their approach only demonstrates effectiveness in towels and ropes, of which areas of interest are easy to find compared with more complex garments. Also, Seita *et al.* [86] proposed a deep imitation learning approach to flatten towels by leveraging RGBD images of towels. They used a FEM-based simulator to simulate the depth and RGB images of towels and used DAgger [87] to train the robot to flatten towels. They found that the RGBD image had the best performance among other image formats. However, towels have a limited state space, indicating that their approach may fail in other garments such as jeans, shirts, sweaters and tshirts.

Apart from learning from human demonstration, some researchers considered robotic demonstrations. Lee *et al.* [88] proposed a reinforcement learning approach to fold towels, where only robot demonstrations are needed rather than human demonstrations and only hour-long demonstrations are required. Features of towels are learned from a fully convolutional network. The policy is a goal-conditioned pick-and-place policy. Their work demonstrated a simplified reinforcement learning approach without excessive sources (human demonstrations) and achieved SOTA towel folding performance (a success rate of 78.3%).

Rather than exclusively using imitation learning, some researchers propose combining reinforcement and imitation learning. Balaguer *et al.*[9] introduced a combined imitation-reinforcement learning approach to folding towels with a ‘momentum fold’, which exploits swinging motions to learn the dynamics of the towels. They demonstrated that imitation learning in reinforcement learning could reduce the search space and converge more quickly than reinforcement-only learning approaches. But imitation learning requires human demonstrations, which have the limitations of requiring a large amount of time and human participants.

Imitation learning with human or robotic demonstrations can enable a robot to learn manipulation skills for deformable objects. However, imitation learning usually requires manual human demonstrations, which are laborious and time-consuming. SOTA lacks research that balances learning effectiveness and the costs of human demonstrations. Also, it lacks research on manipulating complicatedly-structured deformable objects such as garments.

Reinforcement Learning

To avoid laborious human or robotic demonstrations, reinforcement learning in a simulated environment for robotic deformable object manipulation has been widely researched. Robots learn the skills for manipulating deformable objects from simulated environments and transfer the skills in real environments with domain adaptation (DA). Also, robots trained in simulated environments can avoid hazardous actions that may be harmful to human beings and surrounding environments. Quillen and Jang *et al.* [89] proposed a simulated benchmark comparing different reinforcement learning approaches, emphasising off-policy learning and generalisation to unseen objects. They found that reinforcement learning approaches such as the bootstrapped double Q-learning (DQL) [90] are more suitable when the data size is smaller. In contrast, supervised approaches are suitable where data size is plentiful with entire-episode values for supervision. This benchmark directs researchers to choose a proper approach for manipulating deformable objects under different situations and scenarios.

Some researchers focus on rope and towel manipulations of robots with reinforcement learning. Lee *et al.* [91] proposed a force-based reinforcement learning for robotic rope knotting and towel flattening. In their research, forces and gripper poses are transformed into the current scene by a wrapping function computed from non-rigid registration. Then trajectories are derived from variations between demonstrations with feedback gains determining how much to match forces and poses. Their approach features specific forces for specific tasks, improving learned skills' effectiveness. However, in [43], they only tested simple objects such as towels and ropes. Similarly, Matas *et al.* [92] trained a robot to fold a towel and arrange a piece of cloth through a hanger by reinforcement learning. The robot is trained in a simulated environment for around 250 iterations. Training robots in real environments with the proposed reinforcement learning approach is difficult because some actions that can be realised in simulated environments can not be realised in real environments due to the limitations of real robots. Hazardous actions (actions that may harm robots or humans) are easy to handle in simulated environments but must be avoided in real environments. Likewise, Wu and Yan *et al.* [93] introduce a reinforcement-learned-based approach for rope and towel flattening. Unlike traditional pick-and-place approaches, their approach features a placing policy conditioned on random picking points. The relationship between picking and placing is encoded in iterative pick-place action space. However, manipulations took 40 actions for ropes and 80 actions for towels, which is time-costly, and only towels and ropes (simple objects) were tested in their experiments.

Robotic deformable object manipulation with reinforcement learning is also applied in robot-assistive tasks for human beings, such as clothing wearing. Matsubara *et al.* [94] proposed a reinforcement learning paradigm to learn motor skills for wearing a robot's T-shirt. Compared with mainstream research focusing on object states, this research focuses on robotic configurations. The reward signal is topology coordinates, which inform the topological relationship between the configurations of the robot and non-rigid material (T-shirts). The paradigm does not involve the details of robotic and object states; thus the task difficulty was decreased. However, their experiments only tested T-shirts, where learned skills can not be transferred to wearing other garments. The paradigm is restricted to specific garment shapes, indicating that the paradigm is not generic. Meanwhile, Colomé *et al.* [95] proposed a safe-assurance robotic scarf wrapping around a mannequin's neck. This research features a safe wrapping process that can be controlled compliantly using a friction model in a reinforcement learning paradigm. A friction model is a model that detects hysteresis of friction in robotic joints, which causes unsafe actions in learned skills from the reinforcement learning paradigm. Also, the friction model enables track how the robot conducts manipulation strategies.

Twardon *et al.* [15] propose restricting robotic action-space modelling to the actions that are safe for robotic hands and arms, garments and the head when a robot is trained to put a knit cap on a Styrofoam head. A direct policy-search algorithm finds appropriate trajectories in the restricted action space to enable the robotic knit-cap operation. Their approach has demonstrated effective manipulation planning under a restricted action space. Finally, Gao *et al.* [96] proposed a robotic assistive personalised dressing using an iterative path optimisation with vision and force information. Visual information is human poses and the movement space of upper-body joints. Force information is used to detect external force resistance and locally adjust the robot's motion. A synthetic database has been used to train the robot, and their proposed method outperforms SOTA on path errors, computation time and the number of iterations.

Other researchers also applied reinforcement learning for robotic garment and fabric manipulation. Tsumine *et al.* [97] proposed a deep reinforcement learning for robotic cloth manipulation with a smooth policy update. Their networks combine the nature of smooth policy updates in value-function-based reinforcement learning with automatic feature extraction from high-dimensional observations using deep neural networks to enhance the sample efficiency and learning stability with fewer samples. Meanwhile, Jangir *et al.* [98] introduced a deep reinforcement method for dynamic fabric manipulation. Instead of only considering targeted configurations of objects, this approach also considered speed and acceleration that can impact manipulation performance. Also, non-grasped points achieved goal positions by selecting optimal trajectories from grasped points. Their approach revealed the importance of these factors (speeds, accelerations and non-grasped point positions) in robotic deformable object manipulation. However, there are two limitations in their approach: the first limitation is that only fabrics are tested while more complex garments (e.g. jeans, shirts, sweaters, towels and tshirts) are not tested. The second limitation is that they only validated their approach in a simulated environment rather than a real environment. Similarly, McConachie *et al.*[99] proposed a multiarmed bandit approach for deformable object manipulation. Instead of defining specific models for each type of object, their approach features selecting models for manipulation that fit deformable objects from the multiple 'arms'. They balanced explorations of models that fit objects and exploitations of high-utility models. Their approaches solved the limitations in many approaches that the models are only for a specific type of object, but they only tested their approach in simulation environments, which may invalidate real environments.

Predictive models have also been used in reinforcement learning to learn manipulation strategies for deformable objects. Ebert and Finn *et al.* [100] introduced a predictive model in deep reinforcement learning for robotic object manipulation. The predictive model is a self-supervised model that takes the starting and goal configurations of objects as inputs and outputs predicted videos of object motion, which are used in reinforcement learning to learn skills of object manipulation. Their approach works expectedly on rigid objects and simple deformable objects with simple actions such as folding flattened shorts and moving rigid objects. However, the deformations of complex objects (such as garments) are difficult to be predicted due to their almost infinite object space.

Reinforcement learning has been widely applied and researched in robotic deformable object manipulation. However, many approaches are only tested in simulated environments rather than real environments. There exist gaps between simulated and real environments, indicating that success in simulated environments does not guarantee success in real environments. Also, some skills learned in simulated environments cannot be applied in real environments due to the action restrictions of real robots. Ropes, towels and fabrics are researched in these approaches, while other garments such as sweaters, shirts, and jeans are rarely investigated due to their complicated structures. Transferring skills between simulated and real environments is a challenge in the SOTA.

2.3.3 Data-Driven Approaches-Finding Grasping Points

For garments and other complex objects, some researchers chose to find grasping points for manipulating garments rather than model-based approaches or reinforcement and imitation learning approaches. Strategies for finding grasping points include feature detection, grasp and re-grasp, neural networks and matching simulated and real garments. This section summarises previous approaches to finding grasping points for garments and other deformable objects and discusses their limitations.

Feature Detection

A major challenge for model-based and data-driven approaches is that most approaches were not tested on complex deformable objects such as garments. Garments are difficult to model in simulation engines and to collect data in real environments. However, robotic garment manipulation is widely applied in areas such as robot-assistant garment wearing for disabled people and garment sorting. Therefore, approaches for robotic garment manipulations are needed.

Instead of modelling garments in a simulated environment or training robots to learn garment manipulation skills with reinforcement learning, some researchers proposed only finding grasping points rather than simulating entire garments. To flatten or fold garments, only a limited number of grasping points are needed rather than models of entire garments. These approaches usually feature finding grasping points when garments are hanging in the middle of the air and implementing a typical pipeline: finding grasping points, stretching garments, placing garments on tables and folding garments. These approaches are found to be effective for robotic garment manipulation.

These approaches mainly include three kinds to find grasping points: detecting features, grasping and re-grasping, neural networks and matching real and simulated garments. The first approach is about detecting features to find grasping points. Grasping points are particularly wrinkles, corners and folds. Detecting these garment parts is the key to deciding grasping points for manipulating garments.

There are approaches to detecting folds of garments. Stria *et al.* [101] introduced a folding axis detection approach for flattening garments. RGB and depth images were used to find the bottom and top layers of folded garments, where garment surfaces were labelled using an energy minimisation framework (which was trained from a novel garment database) to predict garment poses. Once garment poses were known, candidates of axes for folding were generated, and correct folding axes were selected. One arm of the robot grasped garments from the boundary of the top layers from the selected folding axes, and the other arm pressed the bottom layers in case of slipping. However, their approach is only effective if folding axes are easily detected; thus garments in experiments were flattened with only a part or sleeves folded. Therefore, their approach can not enable garment flattening tasks if garments are crumpled without clear folding axes.

Wrinkles and overlaps can also be used for finding grasping points. Estevez *et al.* [102] introduced a garment-agnostic laundry tasks including garment flattening and ironing. 3D reconstructed garments were analysed in their heights and “bumpiness” values to detect overlapping regions for flattening with an iterative algorithm. For garment ironing, a wrinkleness local descriptor was implemented to detect wrinkles on garments, and the robot controlled its ironing pressure with an iterative path-following control algorithm. Although their approach succeeded in flattening garments, only garments with simple overlaps (without crumpled configurations) are tested, while for most circumstances in daily life, garments are crumpled with irregular overlaps. Manipulating garments with crumple configurations is more difficult and may result in task failure with their approach.

Meanwhile, Triantafyllou *et al.* [103] introduced a vision system to find towel corners by detecting characteristic features of towel corners. Towel corners can then be used to flatten and fold towels, serving as a critical part of a robotic towel manipulation pipeline. A robot threw towels until corners were found. They studied and divided corners: interior and exterior corners, true and pseudo corners, and a-, b-, and c- type corners. The features of corners include junctions of edges and folding axes. They demonstrated an initial and early-staged approach to detecting the grasping points of deformable objects. Similarly, in a more recent work of the authors [104], they introduced a model-based robotic garment flattening pipeline by finding a set of grasping points, detecting corners and outlines of a garment, grasping the garment from the grasping points, placing grasped garment back to the table, matching a "foldable template" with the garment to find another set of grasping points, and finally re-grasping the garment to finish flattening. Their approach features observing characteristics of outlines and junctions of garments to find grasping points. However, their approach requires multiple-grasping of garments and only short-sleeved garments are tested (long-sleeved garments are more difficult to find outlines).

Some researchers suggest that more than one feature can be used to find the grasping points of garments. Triantafyllou *et al.* [105] studied the folding characteristics of folded garments independently of garment shapes. They introduced a hierarchical visual architecture that divides the characteristics into two levels: low-level features, including junctions of edges, and high-level features, including layers and axis of folded garments. These features can later be used to decide on grasping points for robotic manipulation tasks, including flattening and folding. They argued that these features are independent of garment shapes, indicating that this approach is generic for all shapes of garments. Similarly, recently in [106], they find grasping points from garments in their configurations by finding common features in garments. They created a 'dictionary' of features in grasped garments: junctions, edges and folds. By matching features in the 'dictionary' with the features in grasped garments, they have corresponding manipulation strategies to unfold garments. They advanced the state of the art by not considering models for garments or robotic skills for manipulating garments with reinforcement-imitation learning.

Detecting features to find grasping points for manipulating garments demonstrate their effectiveness. However, these approaches are usually only effective for one shape of garments or valid in seen garments. The generalisation of these approaches to multiple shapes and unseen garments is needed in SOTA to ensure that robots can manipulate garments in different circumstances.

Grasping and Re-grasping

Apart from detecting features to find grasping points for manipulating garments, some researchers also proposed grasping-and-re-grasping. Detecting-feature approaches usually use single images from garments. For this, robots will interact with garments (for example, grasping and rotating) and find grasping points until garment configurations (object states) are recognised. For these grasping-and-re-grasping approaches, garments usually hang when their configurations are being recognised, and grasping points are also based on their hanging configurations.

Osawa *et al.* [107] introduced an early development on learning garment hanging configurations to flatten them. Garments were repeatedly re-grasped to predict their shapes and then flattened according to their shape information. Recognition of garment shapes was realised by comparing similarities between grasped garments with each shape. This research demonstrated the performance of priorly learning garment shapes and flattening garments according to their shapes. However, multiple re-grasps are needed for recognising garment shapes, indicating that several iterations and time were needed. Similarly, Cusumano-Towner *et al.* [108] introduced a hidden Markov model (HMM) that was used to recognise garments and track the configurations of manipulated garments. The recognised garments and tracked configurations were combined to plan a garment manipulation. A robot would repeatedly grasp the lowest points of garments until the HMM recognised their "known configurations". Then manipulation strategies would be derived to find grasping points based on recognised "known configurations", and garments would be manipulated. This research was an early-stage development on recognising "known configurations" of garments, but this research requires re-grasps of garments, causing several iterations and time. Also, Maitin-Shepard *et al.*[109] proposed detecting grasping points of grasped towels by observing grasped configurations through dropping and re-grasping. In their experiments, a robot had to find two grasping points of towels, which means multiple dropping and re-grasping are needed. The process is time-consuming for towels, which are simple compared t other garments such as jeans, shirts, sweaters and tshirts.

There are other means of interacting with garments to recognise their configurations and grasping points. Willimon *et al.* [110] used an interactive approach to flatten towels. In their experiments, towels were shaken and flattened by a robot. Shaking means the robot rotates towels from a chosen point with a chosen orientation. Grasping points were chosen based on the features of towels with crumpled configurations, where the configurations of towels were tracked after one manipulation to determine the next grasping point. However, their approach only tested towels and took around 50 "shakings" before they were flattened. "Shaking" more complex deformable objects such as sweaters, shirts, and jeans may take more "shakings".

Meanwhile, Yuba *et al.* [111] proposed a hem/corner detection technique for robotic towel flattening with a "pinch and slide" motion strategy. Crumpled towels were divided into three categories: configurations with one visible corner (State 1), configurations with two visible corners (State 2) and configurations with no visible corner or hem (State 3). If State 1 is observed, pick-and-place actions are conducted to perform flattening. If State 2 is observed, the garment is first picked up from the corner, and the second grasping point on another corner is found from the hung configurations. The following actions are performed to flatten the garment. If State 3 is observed, a shape arrangement manipulation is conducted until State 1 or State 2 is observed. Actions were derived from an operation selection mechanism, a partially observable Markov decision process. Their approach succeeded in flattening towels, while their approach may probably fail in shirts or sweaters. These garments have more complicated object states, thus detecting corners and hems and derivation of actions from the operation selection mechanism are difficult. Finally, Ha *et al.* [112] proposed a "FlingBot" robotic garment flattening pipeline that features a grasp-stretch-fling mechanism. Garments are firstly grasped from a table with crumpled configurations, where grasping points are selected from a value network that evaluates each pixel's "grasping value" on RGB images. Also, fling actions are derived from the value network. Then garments are stretched and flung by a robot to be flattened. The robotic garment flattening pipeline takes advantage of gravity to flatten garments. Gravity can transfer garments from crumpled configurations into simple configurations.

Although grasping-and-re-grasping approaches demonstrate effectiveness in garment manipulations, they usually take a long time to recognise garment configurations because they involve multiple grasps or other actions. The balance between manipulation and operating time is needed for these approaches but it has not been addressed yet in the state of the art.

Neural Networks

Neural networks have also been used for detecting grasping points. Neural networks extract features of garments by being trained in databases containing many garment images. Compared with non-neural-network approaches, neural networks can easily be generalised to garments of different shapes and unseen in the training set.

Qian and Weng *et al.*[113] devised a grasping point detection method for grasped garments by using a network to segment the edges and corners of garments from their depth images. The edges and corners are coloured differently, and a robot will approach the corners to grasp garments. Their approach features an edges and corners detection network, but their approach is limited to towels. Edges and corners can be easily detected in towels, which can be difficult for other garments such as shirts. A deep neural network is preferable for detecting other features, such as wrinkles, to help find grasping points. Similarly, Seita *et al.*[114] introduced detecting grasping points with a deep neural network from bed-making. The network is a YOLO network [115] trained on RGB images. They changed two layers in the YOLO network but kept most of its 32 million parameters. They also trained the network with depth images capturing various bed-sheet configurations with ground-truth grasping points in a transfer learning way. The trained network showed high accuracy in finding grasping points for crumpled bed sheets. However, the network only supports bed sheets because the number of grasping points and bed-sheet configurations is limited. A larger database is needed to manipulate more typed of deformable objects. Gabas *et al.* [53] introduced using three convolutional neural networks (CNNs) to find grasping points for garment grasping tasks. The first CNN was used to predict garment shapes, the second CNN was used to find the first grasping points, and the third CNN was used to find the third grasping point. However, this approach requires rotating garments to predict garment shapes and find grasping points, which is time consuming.

These neural network-based approaches demonstrated their effectiveness in garment segmentation and grasping point localising. However, these approaches usually require a database, which takes additional time to collect. Efficient approaches to collecting databases and training networks can improve the performance of these networks, which should be a research direction in SOTA.

Matching Real and Simulated Garments

Instead of using real data to find grasping points, some researchers proposed matching real and simulated garments to recognise garment configurations and grasping points. Laborious data collection in real garments is not required for these approaches, improving efficiency compared with previously mentioned approaches.

Researchers constructed simulated garments databases, where garments of different shapes and configurations are denoted with pre-designed grasping points. Different matching approaches are applied in experiments to match real garments with simulation garments in databases to find grasping points for real garments.

Kita *et al.* [116] proposed matching 3D information of grasped garments with simulated garments in a database to find grasping points for garments. Simulated garments were constructed using vertices and edges and compared with real garments by overlapping ratios. Although simulated garments are simplified with vertices and edges, found grasping points are not on real garments, causing failed cases. Also, only one shape was tested in their experiments due to the complexity of the garments, because simulating garments was time-consuming and computationally expensive at the time (circa. 2004). Similarly, Li *et al.* [23] proposed to recognise grasped configurations of garments to find grasping points by matching 3D reconstructed mesh models of garments with simulated 3D mesh models in a database. A robot rotates garments to reconstruct their 3D models and re-grasps garments until the 3D models match with 3D models in the database. The grasping points are found from matched 3D models in the database; then, a Baxter robot grasps garments to unfold them. Their approach requires rotations and re-grasps, requiring a certain amount of time. Li *et al.* further improved their approaches in their later project [24]. They advanced their previous approach in [23] by proposing a feature extraction approach to match reconstructed 3D mesh models with the 3D mesh models in the database. However, rotations and re-grasps are still needed for matching, requiring overheads for manipulations.

Similar to the grasping-and-re-grasping approaches, these approaches need grasping and re-grasping to match simulated and real garments, which is time-consuming. There are gaps between real and simulated garments; thus multiple views of garments from grasping and re-grasping are needed to ensure that a robot can match real and simulated garments. Reducing the need for grasping and re-grasping is necessary for SOTA.

2.4 Discussion

This literature review summarises and discusses SOTA robotic perception and manipulation of deformable objects. Robotic perception includes investigating the physics and geometric (shapes) of deformable objects, while robotic manipulation involves model-based and data-driven approaches to deformable objects. Although SOTA demonstrates effectiveness in robotic perception and manipulation of garments, some limitations are not solved.

Most approaches simulate objects in simulation engines with different approaches to investigate the physics properties of deformable objects. However, these approaches do not consider complex deformable objects, such as garments. Garments consist of different components (e.g. pockets, collars, sleeves and buttons) difficult to simulate in existing simulation engines. Even if garments are simulated in these engines, simulation is computationally expensive and time consuming. Measuring physics properties (especially stiffness) directly from real objects is also laborious because specific instruments are needed [11]. SOTA research on various physics properties of deformable objects includes stiffness, elasticity, and materials. These physics properties are essential to understand objects; thus they are important for robotic manipulation. SOTA is still lacking comprehensive research on leveraging existing simulation engines and technologies to investigate the physics properties of garments, where massive computation and long operating times are not required.

SOTA on garment shape recognition mainly consists of two approaches: interacting with garments or detecting features. Interacting with garments to increase the understanding of garment shapes improves recognition success. However, it requires excessive interacting times while detecting features of garments does not have an expected performance on unseen garments. Recently researchers turned their focus to deep networks such as attention networks. Deep networks require a large-scale database to be trained, which is difficult to be collected in a robot-lab environment. Most large-scale databases are collected using online resources such as shopping malls and Google image searches, which are unrealistic scenarios within robotics. Interacting with garments should draw researchers' attention as these approaches usually achieve a high-accuracy performance, but SOTA lacks a balance between operating times and accuracy.

Model-based robotic deformable-object manipulation approaches involve defining models for deformable objects. These models are used for monitoring and predicting object deformations, which are used for manipulation planning. Models are usually constructed with the finite element method, the mass-spring system, point clouds and topological representation. Similar to investigating the physics properties of deformable objects, using these modelling approaches requires massive computation and long processing times. However, researchers used topological representation to simplify deformable-object models, reducing the computation and operating times. But simplifying models means losing objects' feature details, causing robots not to recognise objects under some circumstances. The SOTA lacks the balance between the simplification of models and the feature details of models.

Reinforcement learning and imitation learning have also been explored in the SOTA. Robots learn to manipulate deformable objects in simulated or real environments rather than obtain manipulation strategies from object models. Imitation learning with human or robotic demonstrations makes skill learning faster than reinforcement learning in simulated environments, but it requires laborious human or robotic demonstrations. Some actions learned from simulated environments cannot be realised in real environments due to the limitations of real robots. Gaps between simulated and real environments can also lead to the failures of manipulations. Most reinforcement learning approaches only test on simple-structured deformable objects such as towels, sponges and ropes rather than complex deformable objects such as shirts, sweaters or jeans. Manipulating these complex objects requires a large action space, meaning skill learning can be learned after a long time. SOTA fails to demonstrate reinforcement and imitation learning paradigms in garments (e.g. shirts, sweaters and jeans).

Recent research on grasping points for robotic garment manipulation demonstrates effectiveness and efficiency. Rather than obtaining manipulation strategies from modelling entire garments, these approaches find grasping points from detecting features or applying grasping-and-re-grasping strategies to manipulate garments. There are four approaches for finding grasping points: feature detection, grasping-and-re-grasping, neural networks and matching real and simulated garments. However, current approaches find it difficult to generalise to unseen garments, which is key in robotic garment manipulation. These approaches should use data-driven strategies to ensure their generalisation. Grasping-and-re-grasping approaches require long operating times to recognise garment configurations and find grasping points. Multiple manipulations (rotating or re-grasping) are also needed to match real and simulated garments to recognise garment configurations. SOTA lacks a balance between operating times and manipulation success.

The approaches of robotic perception and manipulation for deformable objects show some degree of success in understanding the physics and geometric (shapes) of deformable objects and manipulating these objects. However, future work is still needed to address the limitations mentioned in this literature review such that robotic deformable-object perception and manipulation can be used in daily life and benefit human society.

2.5 Summary and Advancements of This Research

Section 2.5.1 summarises the limitations in SOTA. Section 2.5.2 introduces how those limitations are addressed in this research.

2.5.1 Summary of the Limitations in SOTA

As a summary of the discussion in section 2.4, these are SOTA limitations:

- The physics properties of complicated structured objects (such as garments) have not been properly investigated in the literature due to massive computation and long processing time to simulate them in simulated engines (in section 2.2.1);
- Although interactive perception demonstrated expected performance in recognising garment shapes, SOTA approaches lack a balance between operating times and recognition

accuracy (in section 2.2.2);

- Although model-based approaches for robotic garment manipulation succeed in using strategies such as topology representation to simplify garments, a balance between simplification and feature details of garments is needed (in section 2.3.1);

- Due to the large action space needed for manipulating garments, using reinforcement and imitation learning to train robots needs a large number of training epochs, which is inefficient (in section 2.3.2);

- Finding grasping points to manipulate garments is proposed and promoted in the literature. However, approaches that require fewer graspings and re-graspings to recognise garment configurations and grasping points are needed (in section 2.3.3).

2.5.2 How the Limitations Are Addressed in This Thesis

To address the limitations in SOTA, this research proposes the following solutions:

- In Chapter 5, this research proposes predicting the physics properties of real garments and fabrics by learning the physics similarities between simulated fabrics. Simulating fabrics does not require massive computations and long processing time, which is more effective than the SOTA;

- In Chapter 4, this research proposes a continuous-perception paradigm to predict shapes and visually perceived weights of garments. This approach features an "early-stop" strategy that balances operating times and recognition accuracy;

- In Chapters 6 and 7, this research develops and implements an effective robotic garment manipulation pipeline. The robot firstly predicts garment shapes. Then the robot recognises the *known configurations* of garments based on their predicted shapes. Finally, the robot will use pre-designed robotic manipulation strategies to manipulate these garments. No grasping-and-re-grasping is needed to recognise the *known configurations* of garments, and no update to manipulation strategies is needed, demonstrating an effective pipeline compared to the SOTA.

Chapter 3

Continuous Perception for Classifying Shapes and Weights of Garments for Robotic Vision Applications

Robotic perception for learning, predicting and classifying physics and geometric properties of garments are crucial for robotic garment manipulation. Understanding these properties helps robots with garment configuration recognition and motion planning. This chapter[1]*, Chapter 4 and Chapter 5 will introduce three approaches to learning geometric (shapes) and physics (area weights, bending stiffness parameters, visually perceived weights) of garments and discuss the potential benefits of learning these properties to robotic garment manipulation as described in Chapters 6 and 7. Specifically, this chapter presents the initial investigation of the continuous perception paradigm to predict shapes and visually perceived weights of garments by implementing a CNN-LSTM network. This chapter considers predicting shapes and visually perceived weights by learning from video sequences rather than single images, which motivates Chapter 4 in devising a garment prediction paradigm by continuously observing video sequences of garments, showing high accuracies on the predictions of shapes and visually perceived weights of unseen garments compared with the SOTA.*

[1] *This chapter has appeared in [17]. Li Duan is the first author and main contributor to this paper.*

3.1 Introduction

This chapter presents an approach to continuous perception for robotic laundry tasks. The assumption is that the visual prediction of a garment's shapes and weights is possible via a neural network that learns the dynamic changes of garments from video sequences. Continuous perception is leveraged during training by inputting consecutive frames, of which the network learns how a garment deforms. To evaluate the hypothesis one in section 1.4, a constructed dataset of 40K RGB and depth images from 200 video sequences were captured while garments were being manipulated. Ablation studies were also conducted to understand whether the neural network learns the physical properties of garments. The results suggest that a modified AlexNet-LSTM architecture has the best classification performance for the garment's shapes and discretised weights. To further provide evidence for continuous perception, the network was evaluated on unseen video sequences and computed the moving average over a sequence of predictions. It was found that the network has a classification accuracy of 48% and 60% for the shapes and weights of garments, respectively.

3.2 Motivation and Objectives

Perception and manipulation in robotics are interactive processes which a robot uses to complete a task [117]. That is, perception informs manipulation while manipulating objects improves the visual understanding of the object. Interactive perception predicates that a robot understands the contents of a scene visually and then acts upon it, i.e. manipulation starts after perception is completed. This chapter departs from the idea of interactive perception and theorises that perception and manipulation run concurrently while executing a task, i.e. the robot perceives the scene and updates the manipulation task continuously (i.e. continuous perception). Continuous perception was demonstrated in a visual deformable object task where a robot must understand how objects deform over time to learn their physical properties and predict the garment's shape and weight.

Due to their materials' physical and geometric properties, garments usually have folds, crumples and holes, which are irregular shaped and configured, making garments to have a high-dimensional state space. Therefore, when a robot manipulates a garment, the deformations of the garment are unpredictable and complex. Due to the high dimensionality of garments and the complexity of scenarios while manipulating garments, previous approaches for predicting categories and physical properties of garments are not robust to continuous deformations [29][3].

Prior research [41][29], [1] has leveraged the use of simulated environments to predict how a garment deforms; however, real-world manipulation scenarios such as grasping, folding and flipping garments are difficult to be simulated because garments can take an infinite number of possible configurations in which a simulation engine may fail to capture. Moreover, simulated environments cannot be fully aligned with the real environment, and a slight perturbation in the real environment will cause simulations to fail.

Garment configurations (garment state space) have high dimensionality. Therefore motion planning for garments requires a high dimensionality space. This motion planning space should represent the dynamic characteristics of garments and the robot's dynamic capabilities to plan a motion successfully. This chapter argues that learning physical and geometric properties is necessary first to enable the robotic manipulation of garments (and other deformable objects). Garment shapes and weights are two important geometric and physical properties of garments. Therefore, this chapter is about learning garments' physical and geometric properties from real-world garment samples. For this, garments are grasped from the ground and then dropped. This simple manipulation scenario allows us to train a neural network to perceive dynamic changes from depth images and learn the physic (weights) and geometric (shapes) properties of garments while being manipulated, see Fig 3.2.

To investigate the continuous perception of deformable objects, a constructed dataset has been captured containing video sequences of RGB and depth images. It is aimed to predict the physical properties (i.e. weights) and categories of garment shapes (geometric properties) from a video sequence. Therefore, the limitations of state-of-the-art were addressed by learning dynamic changes as opposed to static representations of garments [118, 119]. Weight and shape were used as the experimental variables to support the continuous perception hypothesis. It must note that this chapter does not address manipulation since the aim is to understand how to equip a robot to perceive deformable objects visually.

3.3 Materials and Methods

This chapter's hypothesis is that *A robot can predict the shapes and visually perceived weights of unseen garments by implementing a CNN-LSTM network to learn the deformations of grasped garments and making predictions based on moving averages across video frames of grasping garments, where accuracy is at least 10% better than using single images for predictions.* For this, an artificial neural network was implemented that classifies shapes and weights of unseen garments (Fig. 3.2 and Section 3.3.2). The network consists of a feature extraction network,

Table 3.1: MSEs (average and standard deviation) between unseen target features and predicted features

Window sequence size	Mean MSE	Std. Dev. MSE
2 **(depth)**	0.094	0.031
3 **(depth)**	0.084	0.030
4 **(depth)**	0.089	0.030
5 **(depth)**	0.085	0.030

Table 3.2: MSEs (average and standard deviation) between unseen target features and predicted features in weights Classification

Window sequence size	Mean MSE	Std. Dev. MSE
2 **(depth)**	0.048	0.015
3 **(depth)**	0.046	0.015
4 **(depth)**	0.046	0.015
5 **(depth)**	0.047	0.015

an LSTM unit and two classifiers for classifying the shapes and weights of garments. Three consecutive frame images $(t, t+1, t+2)$ were input into the network to predict the shape and weight of the observed garment from a predicted feature latent space at $t+3$. The garment's weight was proposed to be used as an indicator that the network has captured and can interpret the physical properties of garments. Specifically, the garment's weight is a physical property and is directly proportional to the forces applied to the garment's fabric over the influence of gravity.

For the experiments, window sequence sizes for LSTM from 2 to 5 consecutive frames have been considered. The prediction results and the Mean Squared Errors (MSE) of the latent space from target images and the predicted latent space output from the LSTM have been compared. Table 3.3 and Table 3.4 show the results.

Table 3.1 and 3.2 show that the network architecture with a window sequence size of 3 has the lowest MSE. From Table 3.3 and Table 3.4, it can be seen that the neural network with a window sequence size of 3 has a higher prediction accuracy (48.8%) for shapes and weights (52.7%) while comparing to others. However, the window sequence size has little effect on classification and reconstruction performance as the difference in the MSE and classification averages are not statistically significant. Therefore, this chapter chooses a window size of 3 consecutive frames for LSTM.

Table 3.3: Classification accuracy (in percentages) of unseen garment shapes where J is jeans; SH, shirt; SW, sweater; TW, towel; and TS, t-shirt.

Window Size	J	SH	SW	TW	TS	***Average***
2 **(depth)**	43.0	21.0	62.0	57.0	50.0	***46.6***
3 **(depth)**	39.0	23.0	66.0	54.0	62.0	***48.8***
4 **(depth)**	44.0	21.0	0.62	56.0	52.0	***47.0***
5 **(depth)**	47.0	21.0	62.0	57.0	51.0	***47.6***

Table 3.4: Classification accuracy (in percentages) of unseen garment weights

Window Size	Light	Medium	Heavy	***Average***
2 **(depth)**	75.0	11.0	71.0	***52.3***
3 **(depth)**	76.0	12.0	70.0	***52.7***
4 **(depth)**	74.0	12.0	71.0	***52.3***
5 **(depth)**	74.0	12.0	68.0	***51.3***

3.3.1 Continuous Perception Experiment

Garment Dataset To test the hypothesis, 200 videos of garments are grasped from the ground to a random point above the ground 50 cm and then dropped from this point. Each garment has been grasped and dropped down ten times to capture its intrinsic dynamic properties. A height of 50 cm has been chosen because the Xtion camera can capture garments being dropped and grasped well. The lighting is kept constant so that lumination conditions do not affect capturing garment images for training the proposed network. LED bulbs enable the lighting with 500 lighting lux. Videos were captured with an ASUS Xtion Pro, and each video consists of 200 frames (the sampling rate is 30Hz), resulting in 40K RGB and 40K depth images at a resolution of 480×680 pixels.

The constructed dataset [2] in this chapter (shown in Figure 3.1) features 20 different garments of five garment shape categories: jeans, shirts, sweaters, towels and T-shirts. Each shape category contains four unique garments. Garments are made of cotton except for sweaters which are made of acrylic and nylon. To obtain segmentation masks, a green background and a green sweater were used to remove the influence of a human arm. The RGB images were then converted to a HSV colour space and identified an optimal thresholding value in the V component to segment the green background and the arm from the garment.

[2] The constructed dataset can be accessed and downloaded from `https://github.com/cvas-ug/cp-dynamics`

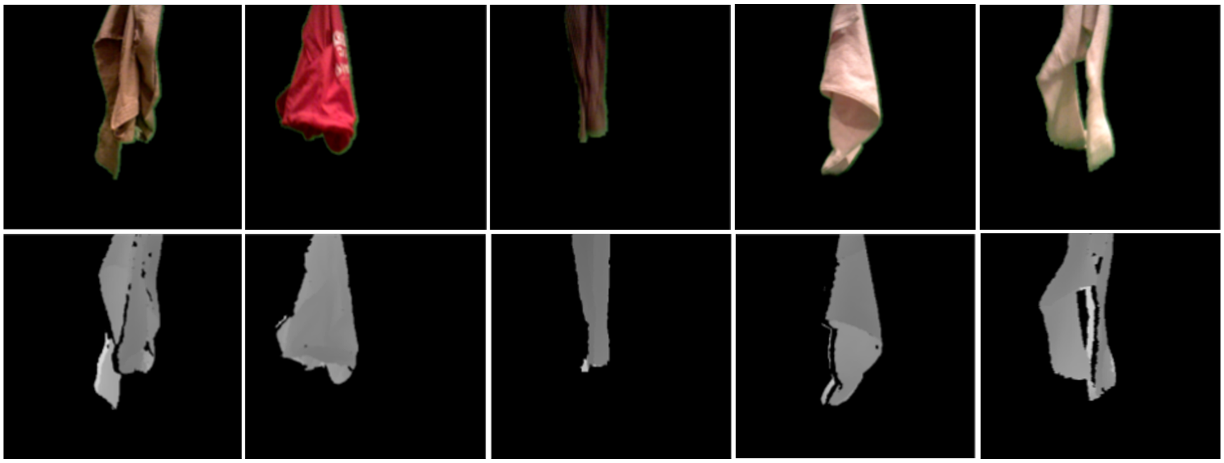


Figure 3.1: *The Constructed Dataset*: The constructed dataset contains five different shapes of garments:(from left to right: shirts, T-shirts, jeans, towels and sweaters), of which RGB images (*top*) and depth images (*bottom*) are captured using an Xtion Pro camera

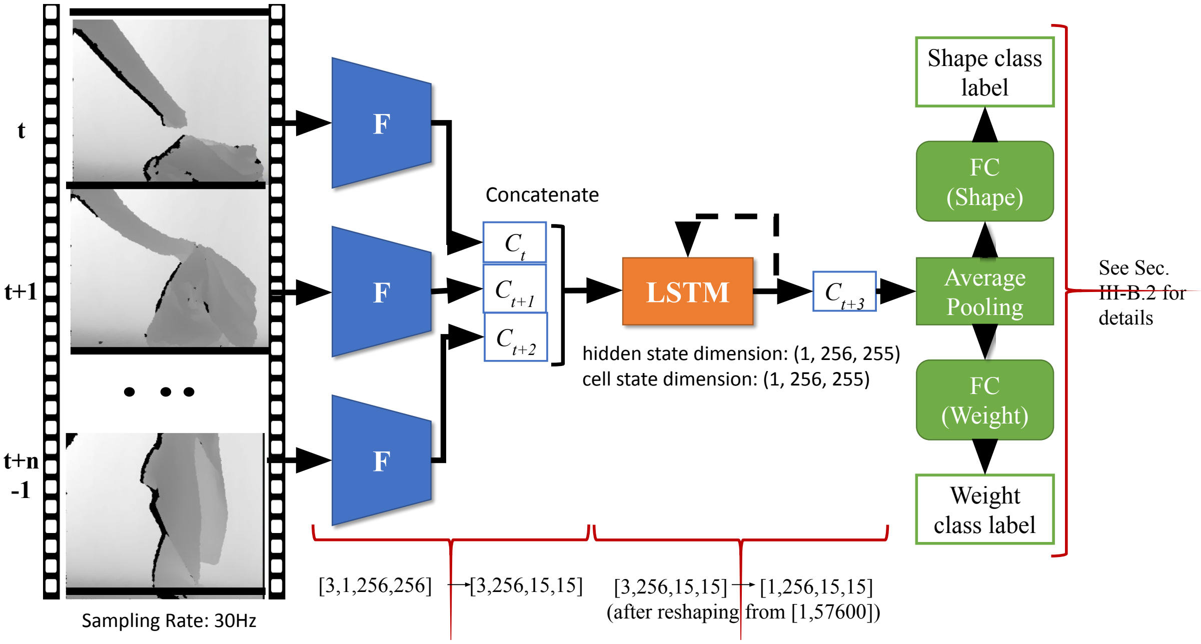


Figure 3.2: The network is divided into feature extraction (F), an LSTM unit and classifier networks. Depth images of a garment with a resolution of 256×256 pixels are passed to the feature extraction network. Three feature latent spaces, i.e. C_t, $C_t + 1$ and $C_t + 2$ from time-steps t, $t+1$ and $t+2$, respectively, are concatenated and then passed to the LSTM. Each feature latent space has a tensor size of 15×15 with a channel size of 256. From the LSTM, a predicted future feature latent space ($C_t + 3$) was obtained, which is reshaped back to the original feature space size (i.e. $[1, 256, 15, 15]$) and input to an average pooling layer. The average pool output with a size of $[1, 256, 6, 6]$ is flattened to $[1, 9216]$ and passed to the fully connected (FC) shape and weight classifiers.

3.3.2 Network Architecture

The ultimate objective is to learn the dynamic properties of garments as they are being manipulated. For this, a neural network comprising a feature extraction network, a recurrent neural network, and shape and weight classifier networks were implemented. Fig. 3.2 depicts the overall neural network architecture. Training this architecture was separated into learning the appearance of the garment in terms of its shape first, then learning the garment's dynamic properties from visual features using a recurrent neural network (i.e. LSTM).

Feature extraction A feature extraction network is needed to describe the visual properties of garments (RGB images) or the topology of garments (depth images). Three state of the art network architectures were therefore implemented, namely AlexNet[120], VGG 16[121] and ResNet 18 [122]. Section 3.4.2 evaluates their performance on extraction features from garments.

Shape and weight classifiers The classifier components in AlexNet, Resnet and VGG-16 networks comprise fully connected layers that are used to predict a class depending on the visual task. In these layers, one fully connected layer is followed by a rectifier and a regulariser, i.e. a ReLu and dropout layers. However, this chapter considers whether the dropout layer will benefit the ability of the neural network to generalise the classification prediction for garments. The reason is that the image dataset used to train these networks contain more than 1000 categories and millions of images [120], while the constructed dataset is considerable smaller (ref. Section 3.3.1). The latter means that the dropout layers may filter out useful features while using the constructed dataset. Dropout layers are useful when training large datasets to avoid overfitting. Therefore, modifying the fully connected networks by removing the ReLu and dropout layers and observing their impact on the shape and weight classification tasks have been experimented. After experimenting with the different network parameters, it was found that the best-performing structure comprises three fully connected layer blocks, each of which contains a linear layer. The number of features stays at 9,216 without any reduction, then the number reduces to 512 in the second layer, and finally, it reduces to 5 for shape and 3 for weight as the outputs of the classifications. These experiments are not included in this chapter as they do not directly test the hypothesis but instead demonstrate how to optimise the classification networks for the shape and weight classifiers in this chapter.

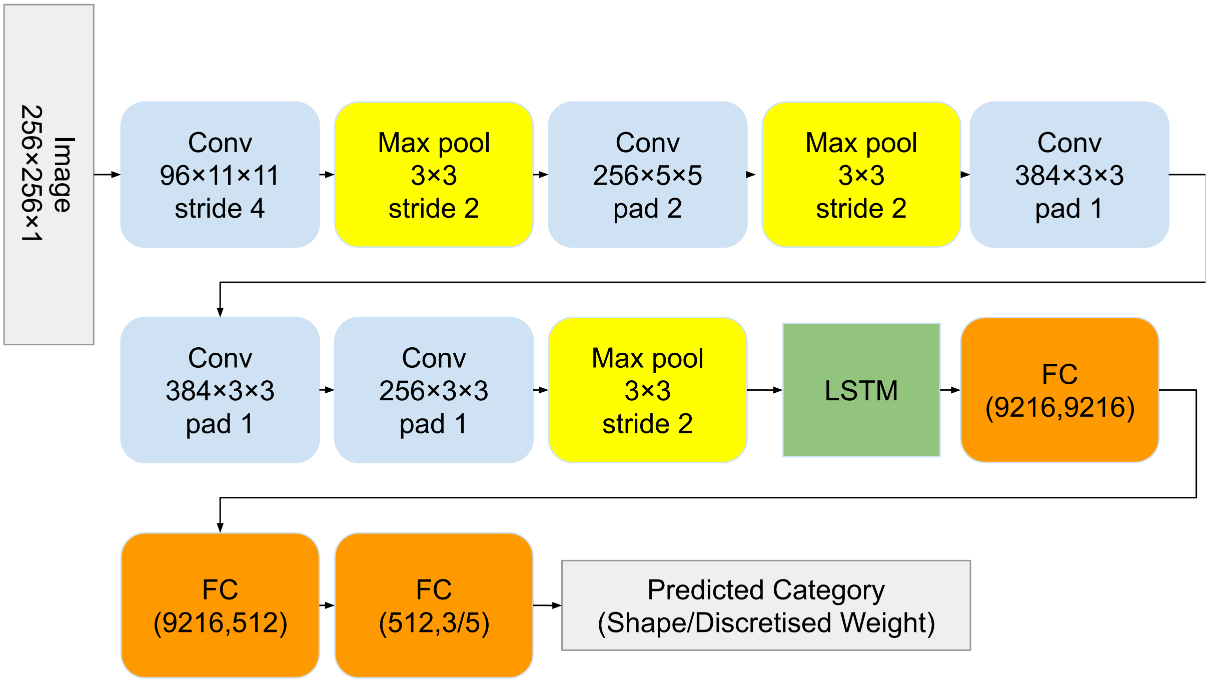


Figure 3.3: *Technical Details of the Proposed Network*

LSTM Rationale The ability to learn dynamic changes in garments is linked to perceiving the object continuously and being able to predict future states. That is, if a robot can predict future changes in garments, it will be able to update a manipulation task on the fly by perceiving a batch of consecutive images rather than receiving a single image and acting sequentially. For this, a Long Short-Term Memory (LSTM) network has been adopted to learn the dynamic changes of consecutive images. After training (ref. Section 3.3.3), the ability to learn garments' dynamic changes was examined by inputting unseen garments images into the trained feature extractor to get their encoded feature maps and inputting these features into the trained LSTM and evaluate if the network (Fig. 3.2) can predict shapes and weights classifications. Figure 3.3 demonstrates the detailed architecture of the proposed network.

3.3.3 Training Strategy

Training (Fig. 3.2) was separated into two parts. First, the network learn the appearance or topology of garments by means of the feature extraction and classification networks (Section 3.3.2). After this, the LSTM network is trained while freezing the parameters of the feature extraction and classification networks to learn the dynamic changes of garments.

Pre-trained architectures have been used for AlexNet, Resnet 18 and VGG 16 but fine-tuned its classifier component. For depth images, the input channel size of the first convolutional layer was changed from 3 to 1 (for AlexNet, Resnet 18 and VGG 16). The loss function adopted is Cross-Entropy between the predicted and target shape labels. After training the feature extraction networks, these networks were used to extract features of consecutive images and concatenate features for the LSTM. The LSTM learning task is to predict the next feature description from the input image sequence, and this predicted feature description is passed to the trained classifier to obtain a predicted shape or discretised weight (visually perceived weight: light, medium and heavy) label. The loss function for training the LSTM consists of the mean square error between the target feature vector and the predicted feature vector generated by the LSTM and the Cross-Entropy between the predicted shape label and the target shape label. The loss function is:

$$L_{total} = L_{MSE} + 1000 \times L_{Cross-Entropy} \tag{3.1}$$

A 'sum' mean squared error has been used during training, but results were reported using the average value of the mean squared error of each point in the feature space. Note that the cross-entropy loss was multiplied by 1000 [3] to balance the influence of the mean squared error and cross-entropy losses.

3.4 Experiments

For a piece of garment, the shape is not an indicator of the garment's physical properties but the garment's weight as it is linked to the material's properties such as stiffness, and damping, to name a few. However, obtaining ground truth for stiffness, damping, etc. requires the use of specialised equipment, and this chapter's goal is to implicitly learn these physical properties. That is, the garment's weight was proposed to be used as a performance measure to validate the approach using unseen samples of garments.

To test the hypothesis, a leave-one-out cross-validation approach has been adopted. That is, in the constructed dataset, there are five shapes of garments: jeans, shirts, sweaters, towels and t-shirts; and for each type, there are four garments (e.g. shirt-1, shirt-2, shirt-3 and shirt-4). Three of the four garments (shirt-1, shirt-2 and shirt-3) are used to train the neural network, and the other (shirt-4) is used to test the neural work (unseen samples). It must be noted that each garment has a different appearance, such as different colours, dimensions, weights and volumes. For weight classification, the garments were divided into three categories: light (the garments

[3] It is found that this value works well with the architecture and database.

Table 3.5: Classification accuracy (in percentages) of unseen garment shapes (Note: J means Jeans, SH means shirts, SW means sweaters, TW means towels, and TS means T-shirts.)

Feature Extractor	J	SH	SW	TW	TS	***Average***
AlexNet(**depth**)	57.0	13.0	71.0	47.0	50.0	***47.6***
AlexNet(**RGB**)	18.0	13.0	0.0	0.0	0.0	***6.2***
VGG16(**depth**)	25.0	18.0	35.0	20.0	25.0	***24.6***
VGG16(**RGB**)	9.0	14.0	20.0	7.0	11.0	***12.2***
ResNet18(**depth**)	6.0	10.0	51.0	69.0	5.0	***28.2***
ResNet18(**RGB**)	16.0	1.0	2.0	81.0	14.0	***22.8***

Table 3.6: Classification accuracy of unseen garment weights.

Feature Extractor	Light	Medium	Heavy	***Average***
AlexNet **(depth)**	72.0	18.0	55.0	***48.3***
AlexNet **(RGB)**	82.0	14.0	7.0	***34.3***
VGG16 **(depth)**	40.0	48.0	31.0	***39.7***
VGG16 **(RGB)**	38.0	3.0	100.0	***47***
ResNet18 **(depth)**	51.0	6.0	47.0	***34.7***
ResNet18 **(RGB)**	41.0	5.0	10.0	***18.7***

weighed less than 180g), medium (the garments weighed between 180g and 300g) and heavy (the garments weighed more than 300g).

A Thinkpad Carbon 6th Generation (CPU: Intel i7-8550U) equipped with an Nvidia GTX 970 has been used, running Ubuntu 18.04. *SGD* was used as the optimiser for training the feature extraction and classification networks, with a learning rate of 1×10^{-3} and a momentum of 0.9 for 35 epochs. An *Adam* optimiser was used for training the LSTM with a learning rate of 1×10^{-4} and a step learning scheduler with a step size of 15 and decay rate of 0.1 for 35 epochs. The reason for adopting different optimisers is that Adam provides a better training result than SGD for the LSTM, while SGD observes faster training for the feature extraction and classifiers. To test the hypothesis, this chapter first experiments on which image representation (RGB or depth images) is the best to capture the intrinsic dynamic properties of garments. Three different feature extraction networks were also experimented with to find the best-performing network for classifying shapes and weights of garments (Section 3.4.1). Finally, the network's performance was evaluated on a continuous perception task (Section 3.4.2).

3.4.1 Feature Extraction Ablation

Three different deep convolutional feature extraction architectures have been tested: AlexNet, VGG 16 and ResNet 18. The performance of shape and weight classification of unseen garments was compared while using RGB and depth images. These feature extractors have been coupled with a classifier without an LSTM, effectively producing single-frame predictions similar to [4].

From Table 3.5, it can be seen that ResNet 18 and VGG 16 overfitted the training dataset. Consequently, their classification performance is below or close to a random prediction, i.e. there are 5 and 3 classes for shape and weight. AlexNet, however, observes a classification performance above chance for depth images. By comparing classification performances between RGB and depth images in Table 3.5, it can be observed that depth images (47.6%) outperform RGB images (7.4%) while using AlexNet. AlexNet performs best among other networks (VGG 16 and ResNet 18). VGG 16 and ResNet 18 are very deep neural networks. But the constructed dataset size in these experiments is small compared with datasets such as ImageNet or CIFAR-10, indicating that VGG 16 or ResNet 18 may be overfitted with this constructed dataset. The reason for depth-map performance is that a depth image is a map that reflects the distances between each pixel and the camera, which can capture the topology of the garment. The latter is similar to the findings in [4, 3].

It can be observed that a similar performance while classifying garments' weights. AlexNet has a classification performance of 48.3% while using depth images. It must be noted that the weights of garments labelled as 'medium' are misclassified as 'heavy' or 'light'. Therefore compared to predicting shapes, predicting weights is more difficult for the neural network on a single-shot perception paradigm. From these experiments, therefore, AlexNet was chosen as the feature extraction network for the remainder of the following experiments.

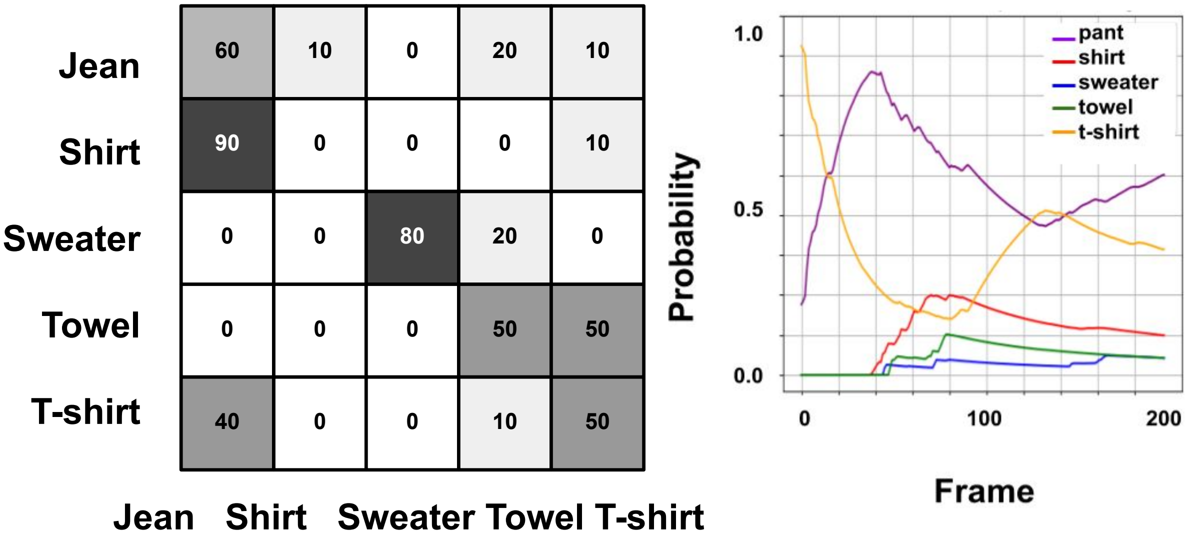
3.4.2 Continuous Perception

To test the continuous perception hypothesis (Section 3.3), AlexNet and a window sequence size of three have been chosen to predict the shapes and weights of unseen garments with video sequences from the constructed dataset, i.e. video sequences that have not been used for training. For this, prediction results were accumulated over the video sequence and a Moving Average (MA) were computed over the evaluated video sequence. MA serves as a decision-making mechanism to decide the shape and weight classes after observing garments deform over the whole video sequence.

This experiment passes three consecutive frames to the network to output a shape and weight class probability for each network output. Their MA values were then computed for each output before sliding into the next three consecutive frames, e.g. slide from frame $t-2, t-1, t$ to frame $t-1, t, t+1$. After sliding across the video sequence and accumulating MA values, an average of the MA values were calculated for each class. The class that observes the maximum MA value was chosen as a prediction of the target category. The unseen test set contains 50 video sequences. Hence, 50 shape and weight predictions were used to calculate confusion matrices in Fig. 3.4 and Fig. 3.5.

From Fig. 3.4(*left*) and Fig. 3.5(*left*), it can been seen that an average prediction accuracy of 48% for shapes and an average prediction of 60% for weights have been obtained for all unseen video sequences. It can be observed in Fig. 3.4(*left*) that the shirt has been wrongly classified as a jean in all its video sequences, but the sweater is labelled correctly in most of its sequences. Half of the towels have been recognised as a t-shirt. Also, for weight, the medium-weighted garments are wrongly classified in all their sequences, where most of them have been categorised as heavy garments, but all heavy garments are correctly classified. Fig. 3.4 (*right*) shows an example of the MA over a video sequence of a shirt. It can be seen that the network changes its prediction between being a t-shirt or a jean while the correct class is a shirt. This is because the shirts, t-shirts and jeans in the constructed dataset are made of cotton. Therefore, these garments have similar physical properties, but different shapes. Meanwhile, Figure 3.6 right shows an example of an unrecognised shirt (which means the shirt is recognised as a pant). The shirt is similar to a pant (in Figure 3.6 left), making it easy for the network to confuse the shirt with a pant. Both shirts and jeans have long-sleeved sides, which mean that they are similar in this chapter. Future work should find an improved neural network (which has been described in Chapter 4 as garment similarity neural network (GarNet)) that can distinguish between jeans and shirts. Fig. 3.5 (*right*) suggests that the network holds a prediction as 'heavy' over a medium-weight garment. This is because heavy garments are sweaters and differ from the rest of the garments in terms of their materials. Therefore, the network can distinguish heavy garments easily.

As opposed to shapes, weights are a physic's property which is difficult to generalise. Nevertheless, the overall performance of the network (48% for shapes and 60% for weights) suggests that the continuous perception hypothesis holds for garments with shapes such as jeans, sweaters, towels, and t-shirts and with weights such as light and heavy, suggesting that further interactions with garments such as in [123, 124] are required to improve the overall classification performance. While validating the network, the overall shape classification performance is approximately 90%, this suggests that the network can successfully predict known garment shapes based on their dynamic properties.



	Jean	Shirt	Sweater	Towel	T-shirt
Jean	60	10	0	20	10
Shirt	90	0	0	0	10
Sweater	0	0	80	20	0
Towel	0	0	0	50	50
T-shirt	40	0	0	10	50

Figure 3.4: Continuous shape prediction (Left: *Moving Average Confusion Matrix*; Right: *Moving Average over a video sequence*)

3.5 Conclusion

From the conducted ablation studies, depth images have a better performance than RGB images because depth captures the garment topology properties of garments. That is, the network was able to learn dynamic changes of the garments and make predictions on unseen garments since depth images have a prediction accuracy of 48% and 60% while classifying shapes and weights accordingly. It is also shown that continuous perception improves classification accuracy. That is, weight classification, which is an indicator of garment physical properties, observes an increase in accuracy from 48.3% to 60% under a continuous perception paradigm. This means that the network can learn physical properties from continuous perception. However, an increase of around 1% (from 47.6% to 48%) was observed while continuously classifying the garment's shape. The marginal improvement while continuously classifying shape indicates that further manipulations, such as flattening [125] and unfolding [126] are required to bring an unknown garment to a state that a robot can recognise. That is, the ability to predict dynamic information of a piece of an unknown garment (or other deformable objects) facilitates robots' efficiency in manipulating it by ensuring how the garment would deform [118, 119]. Therefore, an understanding of the dynamics of garments and other deformable objects can allow robots to accomplish grasping and manipulation tasks with higher dexterity

This chapter has verified this hypothesis:

A robot can predict the shapes and visually perceived weights of unseen garments by implementing a CNN-LSTM network to learn the deformations of grasped garments and making predic-

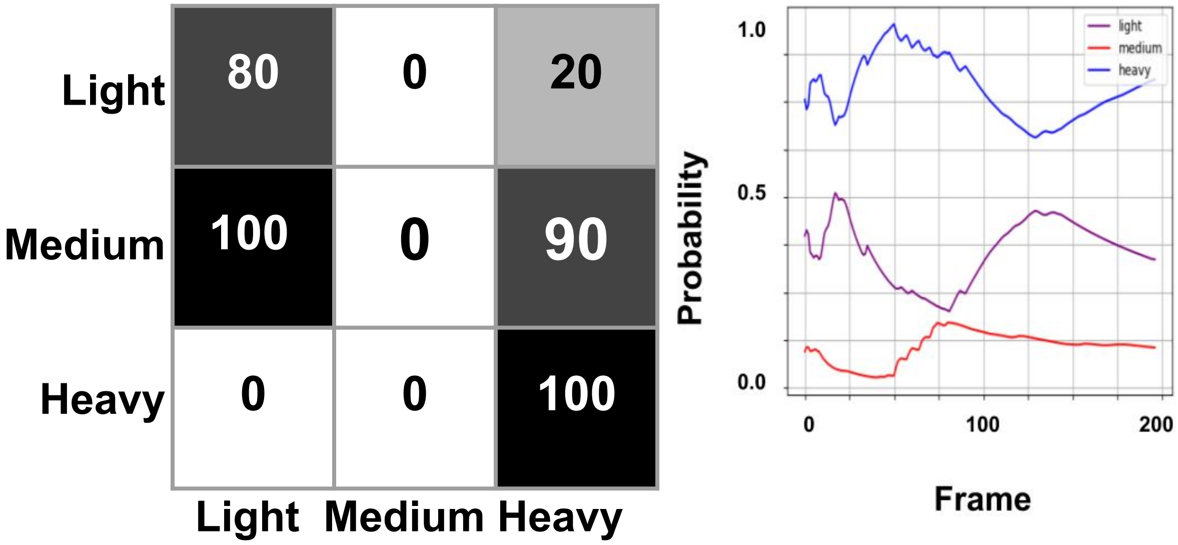


Figure 3.5: Continuous weight prediction (Left: *Moving Average Confusion Matrix*; Right: *Moving Average over a video sequence*)

tions based on moving averages across the video frames of grasping garments, where accuracy is at least 10% better than using single images for predictions.

These results show incorrect classifications of unseen shirts because of their similarity in their materials. Therefore, experimenting on how to improve prediction accuracy on garments with similar materials and structures by allowing a robot to interact with garments as proposed in [124] has been developed and demonstrated in Chapter 4. This chapter also envisages that it can be possible to learn the dynamic physical properties (for example, stiffness) of real garments from training a "physics similarity network" (PhysSNet) (reference: [1]) on simulated data, which has also been developed and demonstrated in Chapter 5.

Although this chapter successfully implements a continuous perception paradigm for predicting shapes and visually perceived (discretised) weights of unseen garments, predictions on garment shapes only slightly improved compared to single images. This chapter adopts a classification strategy, which means the CNN-LSTM network directly outputs predicted categories for unseen garments (shapes or visually perceived weights). Chapter 4 improves the classification strategy by implementing a contrastive learning continuous perception approach, which demonstrates the effectiveness and efficiency of unseen garment predictions.

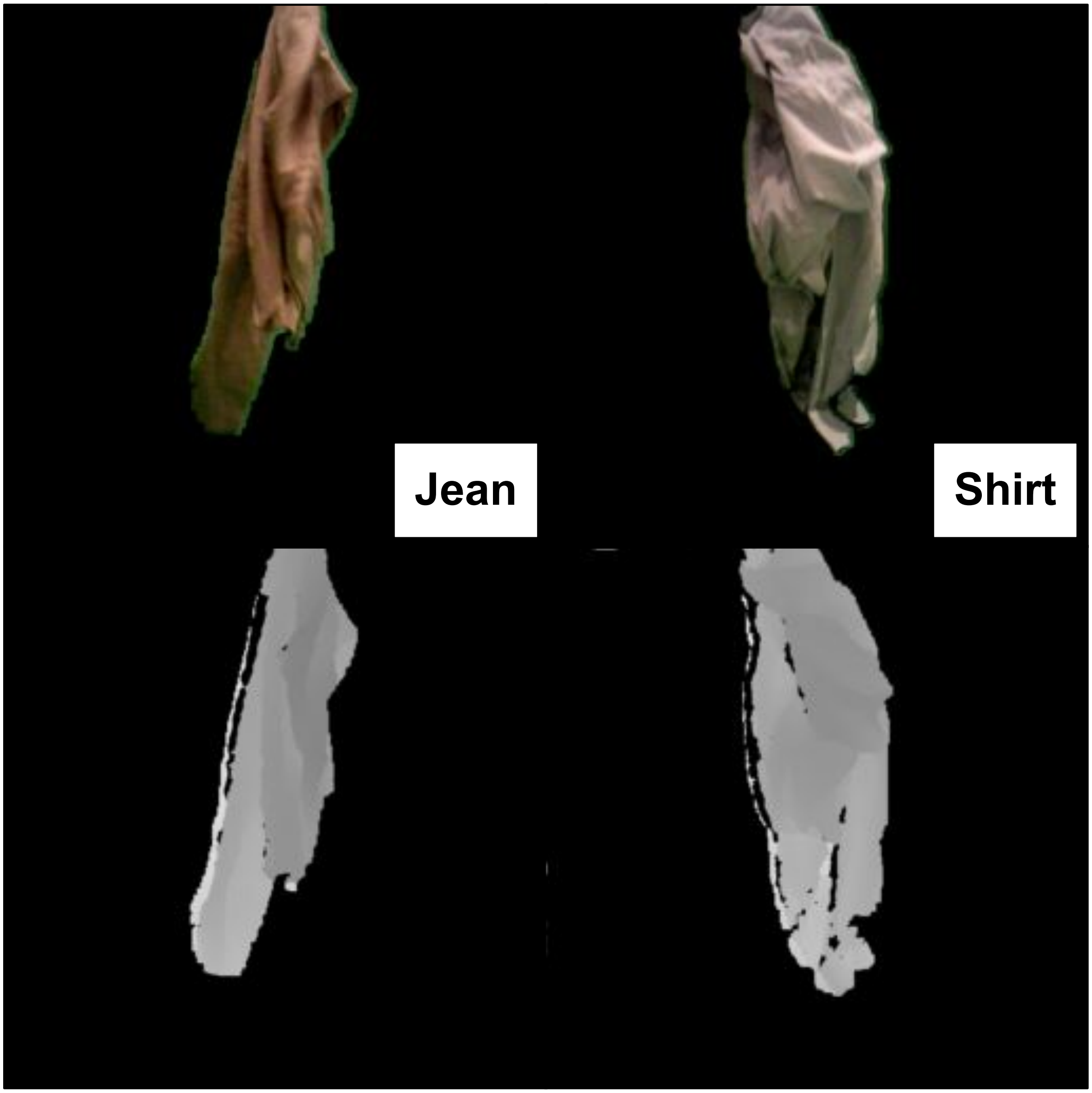


Figure 3.6: *Comparison between a pair of jeans and a shirt*

Chapter 4

A Continuous Robot Vision Approach for Predicting Shapes and Visually Perceived Weights of Garments

Chapter 3 introduced a continuous perception paradigm of "moving average" to predict the shapes and visually perceived weights of garments, indicating that predictions from video sequences of garments being grasped outperform those from single images. However, the results show that the network underperforms. In this chapter [1]*, the continuous-perception mechanism is further developed. A garment similarity network is proposed for learning the geometric similarities between garments of different shapes and visually perceived weights. A robot gains confidence in predicting the shapes and visually perceived weights of garments by continuously perceiving video frames of garments being grasped. A decision will be made when its "confidence" surpasses a threshold. This chapter develops and finalises the research on continuous perception paradigms, revealing its implications on garment shape and visually perceived weight predictions. This chapter also motivates the robotic manipulation part of this thesis in Chapter 7, which is about developing an effective robotic garment pipeline. This pipeline incorporates predicted garment shapes to facilitate robotic garment flattening, reflecting the importance of studying garment shapes in this chapter and Chapter 3.*

This chapter presents a continuous perception approach that learns geometric and physical similarities between garments by continuously observing a garment while a robot picks it up from a table. The aim is to capture and encode a garment's geometric and physical characteristics into a manifold where a decision can be made, such as predicting the garment's shape and its

[1] *This chapter has appeared in [19]. Li Duan is the first author and main contributor to this paper.*

visually perceived weight. The proposed approach features an early stop strategy, which means that a robot does not need to observe a full video sequence of a garment being picked up from a crumpled to a hanging state to make a prediction, taking 8 seconds on average to classify the garment shapes. In the experiments, it is found that the proposed approach achieves prediction accuracies of 93% for shape classification and 98.5% for predicting weights and advances SOTA approaches in similar robotic perception tasks by 22% for shape classification.

4.1 Motivation and Objectives

Compared to previous research focusing on wrinkles [2], and other local features [4, 3], it is proposed to learn the dynamic properties of garments from video sequences and allow a robotic system to recognise the shape and visually perceived weight of a garment continuously. For this, a Garment Similarity Network (termed GarNet in the scope of this chapter) was implemented, which is based on a Siamese neural network architecture that learns the physical similarity between garments to predict shapes (geometric) and visually perceived weights (physical) of unseen garments. A visually perceived weight was defined in three discretised levels using an electric scale to weigh garments physically; namely, light, medium and heavy garment weights. The hypothesises is that *A robot can predict the shapes and visually perceived weights of unseen garments by implementing a garment similarity network to learn the geometric differences between garments and making predictions based on continuously perceiving video frames of grasping garments with an "early-stop" strategy, where accuracy is at least 15% better than SOTA, and the pipeline requires less than 10 seconds.*

To test the above hypothesis, a database has been built that consists of RGBD video sequences of a robot grasping and dropping garments on a table (see Fig. 4.5). This database simulates a sorting scenario (e.g. [4, 2]) where a robot can sort based on shape or weights. This chapter introduces a novel neural network called Garnet (garment similarity neural network), which is trained to learn garments' geometrical and physical similarities based on their shapes and visually perceived weight labels. GarNet's objective is to cluster garments of the same categories (shapes or discretised weights) together and pull garments of different categories apart using a triplet loss function, and these clusters are mapped into a Garment Similarity Map (GSM). To predict unseen garment shapes and weights, the concept of decision points was introduced, which depend on previously mapped points in the GSM. These decision points were used to implement an early-stop strategy by fitting confidence intervals for each cluster to allow the robot to determine whether decision points are within a statistical significant interval around a cluster. Figure 4.1 shows an overview of the proposed approach. The contributions of this chapter are threefold:

1. SOTA has been advanced by adopting a continuous perception paradigm in a neural network which improves the prediction accuracy from 70.8% to 93% for shape classification;
2. The proposed approach can visually estimate weights of unseen garments with a 98.5% prediction accuracy;
3. An early stop strategy is proposed to accelerate inference compared to the SOTA by taking 8 seconds on average to classify shapes.

4.2 GarNet: Garment Similarity Network

The proposed Garment similarity Network (GarNet) consists of a Siamese network, which clusters garments into groups according to their shape and discretised weight categories. In previous work, Siamese networks provided the ability to cluster similar features such as colour features [1] for predicting flag area weights and physics parameters (which is discussed in Chapter 5). Thus, the objective of clustering garments in this chapter is to learn common geometric and physical features of garments of the same categories. The GarNet network comprises a residual convolutional block that extracts features from input data and a fully connected layer that maps features onto a 2D *Garment Similarity Map* (GSM). A garment similarity map is a 2D manifold that encodes a garment's geometric and physical characteristics according to its shape and visually perceived weight categories. Garments of the same categories are clustered together, while garments of different categories are pulled apart. Figure 4.2 shows the garment similarity map in the experiments where each cluster in the map is called a Garment Cluster (*GC*). The GarNet training process is expressed mathematically as $P = f_\theta(I)$, where f_θ denotes a neural network that contains residual convolutional layers and fully connected layers parameterised by the parameters θ, and I denotes an input video frame. P is defined as a garment similarity point (*GSP*). Each frame in the input video sequence of the garments is converted into one garment similarity point (*GSP*). A Garment Similarity Distance (*GSD*) is also defined as $GSD(x,y) = P_i - P_j$, where i and j are the ith and jth garment similarity point. *GSD* increases between garments with different labels and decreases between garments with the same labels. Therefore, to train GarNet, a triplet loss [127] was used:

$$\begin{aligned} PP &= \left|P_{positive} - P_{anchor}\right| \\ NP &= \left|P_{negative} - P_{anchor}\right| \\ TripletLoss &= max(0, PP - NP + margin) \end{aligned} \tag{4.1}$$

where *PP* is a positive pair between positive and anchor samples and *NP* is a negative pair between negative and anchor samples. An anchor sample is an image of a garment. A positive sample is an image of a garment of the same category as the anchor sample. A negative sample is an image of a garment of a different category than the anchor sample. The *margin* is a value that promotes the network to learn to map positive and negative samples further away from each other. The *margin* was set to 1, with reference to [1] where they observed the best performance when setting the *margin* to 1.

Two GarNets are trained separately and predict the shapes and discretised weights independently. That is, the GarNet for shape predictions is trained on shape categories, while GarNet for discretised weight predictions is trained on discretised weight categories, e.g. light, medium and heavy weights. Figure 4.3 demonstrates the technical details of the proposed network.

4.2.1 Garment Cluster Confidence Intervals

To decide which category (either shapes or discretised weights) a mapped garment similarity point in the similarity map belongs to, fitting statistical confidence intervals were proposed to each garment cluster in this map. That is, a confidence interval was defined by using a non-parametric probability density function for each garment cluster, *GC* via a kernel density estimator [128] that is defined as:

$$\begin{aligned} \hat{f}_h(GC) &= \frac{1}{n}\sum_{i=1}^{n} K_h(P - P_i) \\ &= \frac{1}{nk}\sum_{i=1}^{n} K(\frac{P - P_i}{h}) \end{aligned} \tag{4.2}$$

where *GC* is the garment cluster, *K* is a Gaussian kernel, *h*>0 is a smoothing parameter called bandwidth which regulates the amplitude of confidence intervals, and $\hat{f}_h$ is an estimated probability density function for a garment cluster. An ablation study was conducted on confidence interval's bandwidths (*h*) and results are presented in section 4.4.2. After training a GarNet, the centroid of each garment cluster is defined as:

$$GC_{mean} = (\frac{1}{m}\sum_{i=1}^{m} x_{P_i}, \frac{1}{m}\sum_{i=1}^{m} y_{P_i}) \tag{4.3}$$

where GC_{mean} is the mean value of garment similarity points mapped from one garment cluster (in Figure 4.2), and m is the number of garment similarity points in the cluster.

In the experiments, unseen image frames of garments acquired by the robot were directly input to GarNet to map them into the garment similarity map. To decide the shapes and visually perceived weights, a decision point (*DP*) was defined as the mean value of garment similarity points (*GSP*s):

$$DP = (\frac{1}{n}\sum_{i=1}^{n} x_{P_i}, \frac{1}{n}\sum_{i=1}^{n} y_{P_i}) \tag{4.4}$$

where n is the total number of frames observed. To predict the shapes and visually perceived weights, if a *DP* is within any confidence interval and has the minimum distance to the confidence interval's GC_{mean}, a prediction is done. For this chapter, Euclidean distances were used to evaluate how close a *DP* is with respect to GC_{mean}.

Each video sequence has 60 frames (6 seconds: The frequency (10Hz) is lower than that in Chapter 3 because a lower frequency allows more time for the garment similarity network (GarNet) to process images.). Therefore, there are 60 decision points. To predict shapes and visually perceived weights, a predicted category should have at least 80% of decision points belonging to a garment cluster. If none of the categories fulfils this requirement, the observed garment is set as not having a known class. That is, if a decision point is outside any confidence interval, the network is not confident about which category the input garment belongs to. By clustering garments and defining confidence intervals, it is possible to define an early-stop strategy to allow a robotic system to stop its execution if it is confident about the garment shape or visually perceived weight. After observing a number of image frames of a garment (20 images), if any of the trained categories takes 80% of the decision points, the process is terminated, and the category is chosen as the predicted category.

4.3 Experiments

4.3.1 GarNet Architecture

A GarNet comprises a ResNet18 [122] for a feature extraction and fully connected networks (FC). The FC networks comprise three linear layers, where a PReLU activation layer is placed between adjacent linear layers. An Intel-i7 equipped with an Nvidia GTX 1080 Ti was used to train the network. An Adam optimizer with an initial learning rate of 10^{-3} was used, controlled by a learning scheduler with a decay rate of 10^{-1} and a step size of 8 epochs. The network is trained for 270K iterations with a batch size of 28, taking approximately 30 minutes.

4.3.2 Data Collection and Experiments

The video database in this experiment consists of 20 garments of five different shapes, namely, jeans, shirts, sweaters, towels and t-shirts. Figure 4.4 shows garment samples for each garment instance in the database, i.e. five categories and four garments for each category. For each shape, there are four garments of different colours and materials. An electric scale was used to weigh every garment and divide its weights into three discretised levels (called visually perceived weights), namely, light, medium and heavy weights. Therefore, these experiments do not predict the real weight values of tested garments but the discretised weights levels to enable a robot to sort garments. One GarNet for shape and one for visually perceived weights were trained. This is because each encodes knowledge based on the observed features, which result in different, uncorrelated 2D manifolds, as can be observed in Fig. 4.2. For example, jeans and sweaters (heavy weights) are close together in Fig. 4.2(left) but are encoded into heavy in Fig. 4.2(right), which does not correlate to the shape manifold if their clusters are merged.

To validate the proposed network and test the hypothesis, a leave-one-out cross-validation methodology was proposed. That is, all garments were grouped into four groups, and each garment category, as shown in Fig. 4.4, has four different garment instances. Hence, four experiments were conducted, where three groups served as training groups, and one group served as a testing group. The testing group only contains image frames of unseen garments, which means these images are not included in the training groups. It is ensured that the garments in the testing group are entirely 'new' and 'unseen' to the robot. Accuracies were averaged for each category output from the four experiments and used the testing group to validate the classification performance. For each of the four experiments, the training group represents 80% of the video sequence database, while the testing group represents 20% of the database.

A Baxter robot was used to manipulate garments. The Baxter robot grasped garments from a fixed point, lifted the garments to a point above the table (height is 1m) and then dropped the garments to fall on the table. The running time is 6 seconds where the robot grasps a garment and stops in the air for 2 seconds before dropping the garments off to the table. An Xtion depth-sensing camera is used to capture garment video sequences. Each garment is captured ten times, meaning the grasp-and-drop operation is conducted ten times. There are 200 videos in total, and each video contains 60 frames (sampling frequency is 10Hz; video sequence length is 6 seconds). Therefore there are 12,000 image frames in total. Figure 4.5 shows the experimental setup of the robot grasping and dropping garments.

The experiments include 50 unseen garment videos containing ten videos for each of the four leave-one-out cross-validation experiments. For each video sequence, the shape and visually perceived weight of the garment were predicted in the video. Therefore, there are ten predictions for each category (one prediction for each video) and 50 predictions in total. The prediction accuracy for each category is defined as the percentage of correctly predicted videos.

The proposed approach was compared with four SOTA (SOA). Chapter 3 proposed classifying shapes and visually perceived weights by leveraging a convolutional neural network and a long-term short-memory unit (CNN-LSTM). Sun *et al.* [2] provided an interactive approach to classifying garments based on a multi-class Gaussian-Process classifier where a robot gains confidence in predicting garment shapes by shaking and flipping the garments. In their later project [4], they propose to classify garment shapes with a global-local-features classifier, where the classifier captures two local features: local B-Spline Patch (BSP) and locality-constrained linear coding, and three global features: Histogram of Shape Index (SI), Histogram of Topology Spatial Distances (TSD), and Histogram of Local Binary Pattern (LBP). Martinez *et al.* [3] introduced a continuous perception method to classify the shapes of garments, where a robot observes video sequences being grasped and dropped and makes decisions on garment shapes based on these video sequences. The proposed approach was compared with these four approaches on shape classification accuracy and running time if reported.

4.4 Results

While training GarNet, the proposed approach achieves a validation classification performance of 93.9% for shapes and 94.9% for visually perceived weights. Figure 4.2 shows the mappings of testing garments onto the similarity map, where it is possible to observe that garments of different categories are pulled apart; while garments of the same categories are clustered together. These results confirm that GarNet coupled with a triplet loss function (Eq. 4.1) is able to extract physical dissimilarities between categories while maintaining inter-class physical properties within well-defined clusters.

4.4.1 Continuous Perception Experiments

Examples of the experimental results are shown in Figure 4.6. It can be seen that although GarNet does not recognise garments correctly from the beginning (because most of the decision points belong to an incorrect category), GarNet gradually gains confidence in predicting a correct category for each garment because more decision points are within the correct category and percentages of decision points are eventually over 80%. The classification task is consequently stopped early, and the system does not need to observe the full video sequence to make a correct prediction of the garment class. Two ablation studies were conducted for the continuous perception experiment. The first study compares predictions only on local garment similarity points (*GSP*s) rather than on decision points (*DP*s). The second ablation study compares the performance of GarNet trained on RGB and depth images. Tables 4.1 and 4.2 show the results of the leave-one-out cross-validation experiments, where the network achieved 93% for shape classification and 98.5% for visually perceived weight classification. The results show that the proposed network is able to classify shapes and visually perceived weights of unseen garments.

Decision points, *DP*s, (Eq. 4.4) were used to predict unseen garment shapes and visually perceived weights. That is, the position of a decision point on the garment similarity map (in Figure 4.2) depends on all previously observed image frames rather than on currently observed image frames. From Tables 4.1 and 4.2, It can be observed that using decision points has better performance than using garment similarity points (93% vs 78% for shapes and 98.5% vs 80% for visually perceived weights, respectively). This shows that GarNet benefits from using accumulated knowledge via decision points rather than episodic knowledge as in [51, 4].

To investigate whether the type of image affects the overall prediction of a garment class, Three GarNets were trained using RGB, depth and RGBD images. Tables 4.1 and 4.2 show that a GarNet trained on depth images outperforms a GarNet trained on RGB images and a GarNet trained on RGBD images (93% vs 53.5% vs 47.5% for shapes and 98.5% vs 65% vs 58.5% for visually perceived weights, respectively). The increase in performance is because depth images capture structural and dynamic information of the garment being manipulated and are better suited to capture the physical properties of garments as opposed to RGB images as proposed by

Table 4.1: Table: Prediction Results (shapes)

Category	depth, DP	depth, GSP	RGB, DP	RGB, GSP	RGBD, DP	RGBD, GSP
jeans	97.5%	82.5%	97.5%	87.5%	57.5%	10.0%
shirts	77.5%	75%	87.5%	62.5%	55.0%	80.0%
sweaters	97.5%	85%	25%	20.0%	60.0%	5.0%
towels	92.5%	65%	50%	17.5%	17.5%	0.0%
t-shirts	100%	82.5%	32.5%	22.5%	47.5%	37.5%
Average	**93.0%**	78.0%	53.5%	42%	47.5%	26.5%
SD	8.1%	**7.3%**	29.5%	28.1%	15.6%	29.7%

Table 4.2: Prediction Results (visually perceived weights)

Category	depth, DP	depth, GSP	RGB, DP	RGB, GSP	RGBD, DP	RGBD, GSP
lights	97.5%	50%	37.5%	5.0%	90.0%	55.0%
mediums	97.5%	87.5%	72.5%	58.75%	48.75%	65.0%
heavies	100%	87.5%	71.25%	71.25%	52.5%	18.75%
Average	**98.5%**	80.0%	65.0%	53.0%	58.5%	44.5%
SD	**1.2%**	15.0%	13.8%	24.6%	15.8%	21.3%

[1]. This phenomenon has been explored in Chapter 5 and found that RGB images capture visual texture information of garments, but this information is affected by lighting conditions that vary between experiments, resulting in worse performance than depth images. Furthermore, texture information from the RGB images is not constant because accessories of garments, colours and lighting conditions quickly change across different garments. It can be observed that the GarNet trained on the RGBD dataset unperformed compared to only training on depth or RGB images in Tables 4.1 and 4.2. According to Bui *et al.* [129]'s study, grayscaled RGB images outperformed original RGB images as grayscale information has more discriminative powers of extracted features and can be earlier processed in convolutional networks than RGB images. Therefore, this chapter utilised grayscaled RGB images for training and testing GarNet. In Shaikh *et al.* [130]'s work, the challenge existing in RGBD images is that each modality (RGB and depth) is different in temporal and spatial information. The information stored in each modality has different properties, suggesting that RGBD images can potentially underperform compared with depth images. This experiment demonstrates that depth images can better capture the geometric properties of garments and train a GarNet with a high accuracy in predicting the shape and visually perceived weight categories. In this experiment, RGB-D images are obtained by concatenating (stacking) the channels of RGB and depth images together and used to train a GarNet. The concatenation (stacking) causes different modalities in RGB and depth images to affect each other. Future work can implement learning two modalities (RGB and depth) separately using two backbones and concatenating their latent features together to investigate whether training accuracy can be improved.

Table 4.3: Bandwidth Ablation Study.

Bandwidth	Shapes	visually perceived weights
10%	2.0%	4.0%
25%	16.0%	26.5 %
50%	46.0%	26.5%
75%	84.5%	79.5%
95%	**93.0%**	98.5%
99%	82.0%	**99.0%**

Note that the intra-class variability for the jeans category is consistent (i.e. jeans have been used for this category, see Fig. 4.4, top row). Therefore, classification scores in Table 4.1 for jeans are high across the ablation studies with respect to other shape categories with high intra-class variability. This result shows that in order to generalise to unseen garments, depth images and decision points offer the best combination for the continuous perception task.

4.4.2 Ablation study on the confidence intervals bandwidths

A bandwidth, as defined in section 4.2.1, determines the size of a confidence interval. Therefore, the effect of the bandwidth selection was evaluated with respect to the performance of GarNet. A confidence interval of a garment cluster is a region in the garment similarity map of which a certain percentage of *GSP*s are grouped.

A decrease in the bandwidth value denotes a decrease in the percentage of *GSP*s included within the garment cluster. An increase in the bandwidth means that almost all *GSP*s should be included, while a small portion of points is relatively far away from a cluster. This means that a confidence interval may overlap with other confidence intervals, or multiple confidence intervals will be generated for one garment cluster. The final classification prediction depends on the bandwidth.

In Table 4.3, it is found that a bandwidth of 95% has the best performance, so 95% as the bandwidth was used for the rest of the experiments. However, at a 99% bandwidth, it can be observed that the prediction accuracy drops while classifying shapes because *GSP*s are grouped into incorrect categories.

4.4.3 Comparison with the SOTA methods

In these comparisons, due to different research approaches and experiment settings, the different database has been used. However, this chapter used the same garments in the compared approaches ([17], [2], [4] and [3]) to demonstrate that the comparison is fair and gives an idea of how different approaches performed on the same garments. As observed in Table 4.4, GarNet outperforms previous work on predicting unseen garment shapes. Also, compared with Chapter 3 (17 seconds) and [4] (180 seconds), the proposed approach is also faster (8 seconds) because the robot continuously perceives garments without interruptions. There are several reasons why the proposed approach has the best performance.

The use of a garment similarity map to encode knowledge of garment shapes and weights

Inspired by [1], where the authors proposed learning the physical similarity between simulated fabrics and predicting the physics properties of real fabrics, it is found that the similarity network effectively predicts unseen deformable objects, such as garments and fabrics. In Chapter 3, where the main focus was on utilising solely the classification approach rather than the clustering approach, it is found that the clustering approach presented in this chapter improves over the classification approach. Compared with a traditional classifier that regresses embeddings of data into labels (which is equivalent to asking which shape/visually perceived weight classes the data belongs to), GarNet learns geometric and physical characteristics that makes them the same or different (which is equivalent to asking why the data presents the same or different shapes/visually perceived weights). Therefore, for unseen garments, the network only needs to decide the similarities of the garments for each garment cluster rather than classify them into certain classes, reducing the prediction difficulty.

Continuous Perception.

Traditional methods such as [4, 51] that predict shapes and weights of garments are based on static garment features such as wrinkles, outlines, and creases, to name a few. Instead, the GarNet experiments propose to carry out predictions on encoded knowledge in the *GSM*s while learning the dynamic properties of garments.

Table 4.4: Comparisons with the SOTA. The running time is given in seconds and *NA* means Not Available

Method	Accuracy (%)	Time
CNN-LSTM (classification) [17]	48%	17
Interactive Perception[2]	64.2%	*NA*
Single-shot category recognition[4]	67.0%	180
Continuous Perception in [3]	70.8%	6
GarNet (Continuous Perception)	**93.0%**	**8**

Early-Stop strategy.

Compared with [2][3], where the proposed approaches consist of having a robot observe the entire interaction with a garment, the proposed approach only requires a robot to observe interactions partly if termination requirements are satisfied. Therefore, the proposed approach has a mechanism to stop manipulation on the fly, as GarNet can process images every 0.1 seconds, taking an average of 8 seconds to generate a prediction.

4.5 Conclusion

This chapter presented a garment similarity network (GarNet) that learns the similarity of the garments and continuously predicts their shapes and visually perceived weights. A garment similarity map (*GSM*) was also introduced that encodes garment shapes and visually perceived weights knowledge into clusters. These clusters were then used to decide which cluster unseen garment samples belong to heuristically. The experimental validation shows that GarNet obtains high prediction accuracies while classifying shapes (93%) and visually perceived weights (98.5%), Fig. 4.6. Similarly, GarNet's performance was compared with SOTA, where GarNet showed an increase of 22.2% in classification accuracy performance (Table 4.4).

Compared with previous work on continuous perception [3], GarNet has the advantage of an 'early stop' strategy. That is, a robot does not need to observe the full motion (video sequences) to make predictions, enabling robots to be more responsive and effective while manipulating garments and deformable objects in a laundry pipeline. However, GarNet, in this chapter, does not support online learning of unknown garment shapes. For instance, GarNet was trained on five shape categories, and it can predict the shapes of unseen garments from those categories. Enabling a robot to recognise garments of unknown categories is pivotal in future work. Currently, the approach only supports classifying garments into known categories. To extend Garnet to unknown categories, implementing a novelty detection approach and using the GSM to detect whether the observed garment is unknown based on the distance from the known clusters is proposed. Then, a continual learning approach (e.g. [131]) can be adopted to allow GarNet to be retrained without losing previous knowledge.

In future work, devising an online-learning approach is planned for GarNet to investigate complex manipulation interactions (enabled by a behaviour-based reinforcement learning agent [132]), such as twisting garments, shaking garments or rotating garments. From those interactions, differences in stretching and bending characteristics of garments can be exploited to evaluate garments' stiffness parameters, which can potentially help to develop a robot dexterous garment manipulation approach for folding [126], flattening [133], to name a few, that requires fewer iterations. Furthermore, knowledge of the garment's weights can enable a robot to plan for these complex manipulation interactions since it will be possible to reduce the search space while estimating the dynamic physical properties of garments.

This chapter has verified the following hypothesis: A robot can predict the shapes and visually perceived weights of unseen garments by implementing a Siamese network to learn the geometric similarities between garments and making predictions based on continuously perceiving the video frames of grasping garments with an "early-stop" strategy, where accuracy is at least 15% better than SOTA, and the pipeline requires less than 10 seconds.

Compared with Chapter 3, this chapter emphasises introducing a garment similarity map to implement a continuous perception paradigm and features an "early-stop" strategy. The results reveal that this chapter's contrastive-learning continuous perception approach outperforms the classification-based continuous perception approach in Chapter 3. This chapter also demonstrates that a robot does not need to observe the whole video sequence of unseen garments being grasped to make decisions on shape/visually perceived weight categories, balancing the time taken and accuracy in the continuous perception paradigm. Prior knowledge of garment attributions can facilitate a robot to conduct garment manipulations. Garment shape prediction introduced in this chapter contributes to an effective robotic garment flattening pipeline illustrated in Chapter 7. However, other garment attributes, such as bending stiffness, area weights, and other physics properties, are also essential for such a pipeline. These physics properties are usually difficult to be measured in reality due to the lack of specific instruments [11], but they are easy to obtain from simulated environments. Chapter 5 introduces predicting physics properties of real garments and fabrics by learning physics similarities between simulated fabrics, which demonstrates that predicting of physics properties of real and complicatedly-structured objects, such as garments, can be achieved by learning physics properties of simulated and simply-structured objects, such as fabrics.

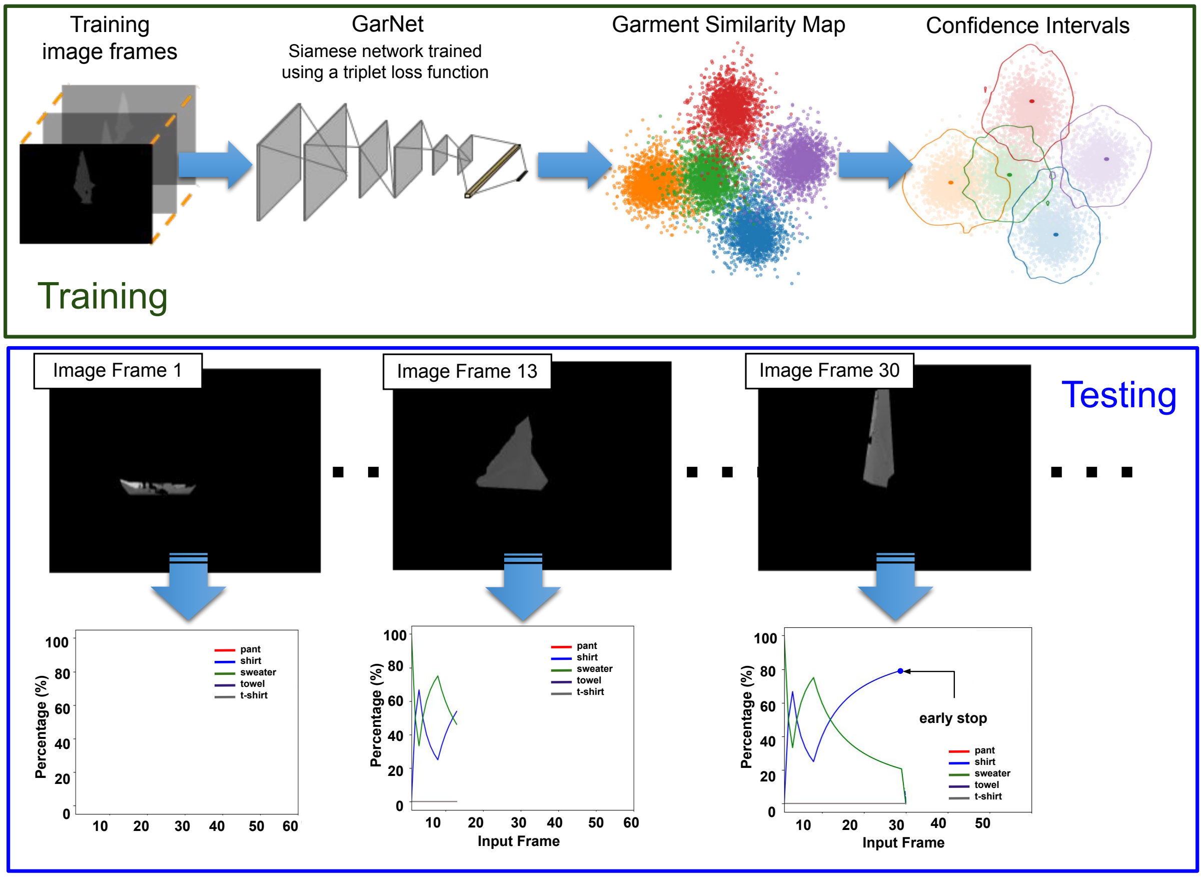


Figure 4.1: *Top: Training GarNet.* A positive, negative and anchor image samples from a video sequence of the training dataset are input into GarNet. Training consists of identifying whether any two of the input triplet comes from the same shape or discretised weight categories. GarNet maps input image frames into a Garment Similarity Map (GSM) in which input frames are mapped into clusters if they are similar; otherwise, new points are pulled apart from the cluster. Confidence intervals are computed for each cluster in the GSM as described in Section 4.2.1. *Bottom: Continuous Perception (Testing GarNet).* An image from a video sequence of the testing dataset is input into a trained GarNet to get the mapping onto the garment similarity map. A video sequence of a garment in the database contains 60 frames. The plots show that GarNet gains confidence in predicting that the perceived garment is a shirt. That is, as knowledge is being accumulated into the GSM, most of the decision points belong to the shirt category. In the example shown, the prediction is stopped at frame 30 because 80% of all decision points belong to the shirt category. The length of video sequences of the robot moving from the table to the top of the lift is about 40 frames.

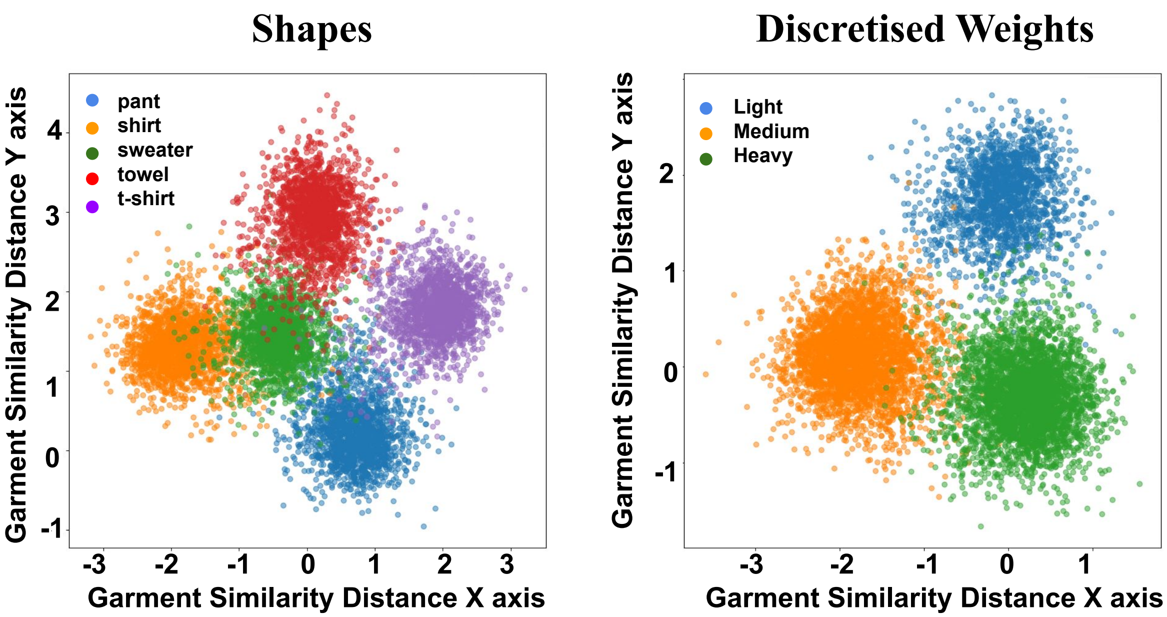


Figure 4.2: The Garment Similarity Map (GSM) after training GarNet.

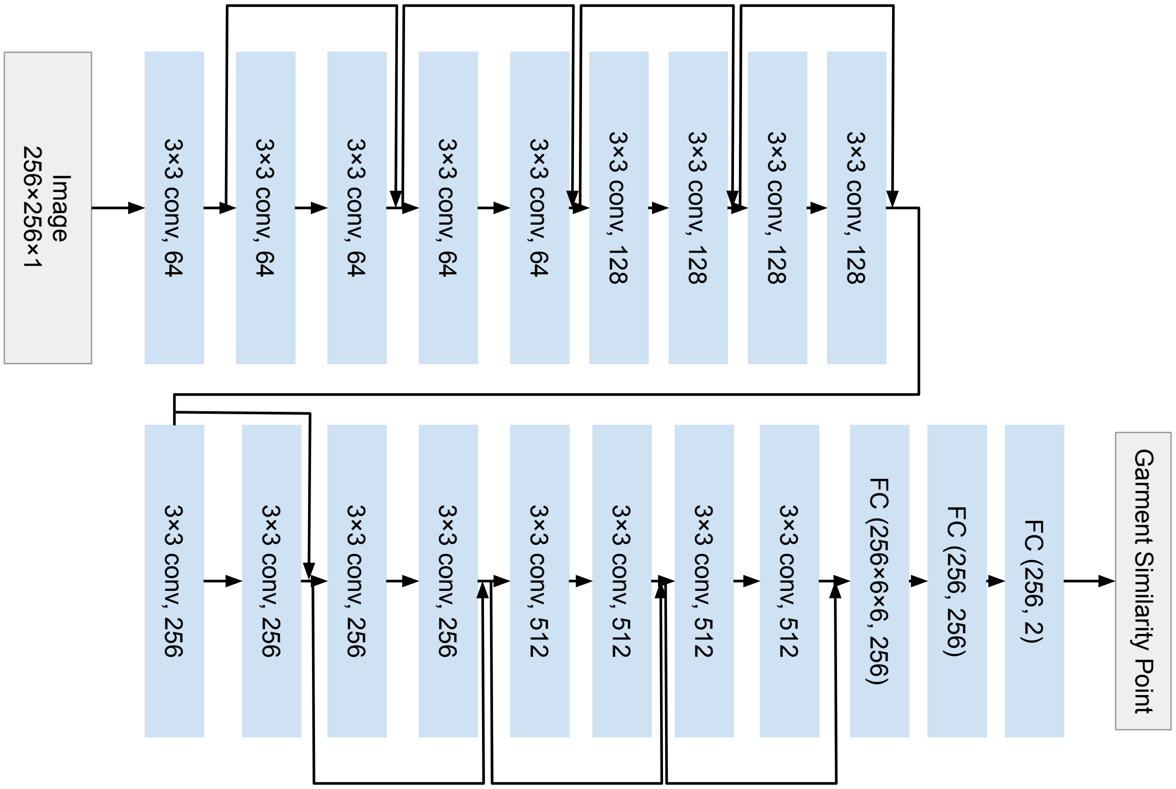


Figure 4.3: *The technical details of the proposed network*

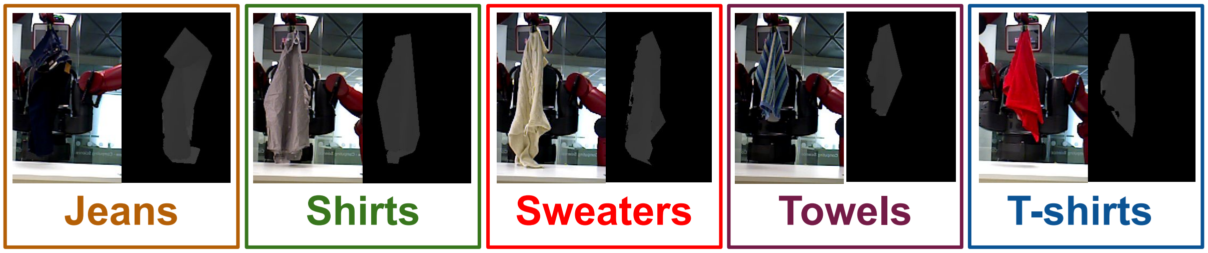


Figure 4.4: *The garment database used in the experiments*: Five categories (jeans, shirts, sweaters, towels and t-shirts), and each category has four garment instances. For each garment instance, an RGB image frame and its corresponding segmented depth image are shown.

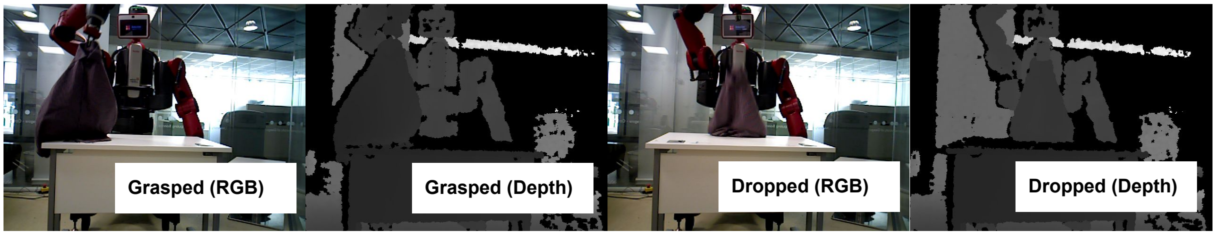


Figure 4.5: An example of the experimental setup for capturing a database of garment deformations. That is, a dual-armed robot grasps garments from a pre-defined fixed point and then drops it. An Xtion camera is placed in the front of the robot to record a video sequence of RGB and depth images.

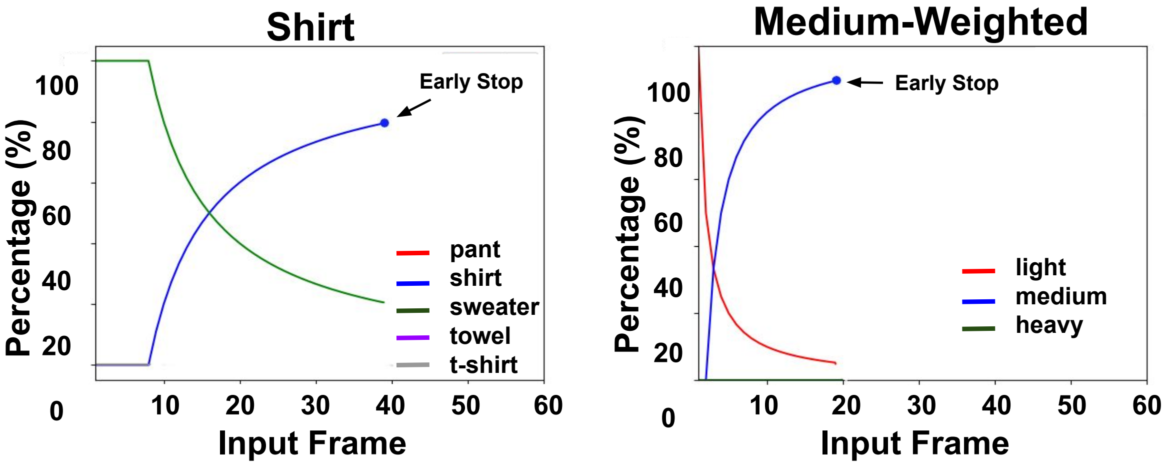


Figure 4.6: Examples of the early-stop strategy proposed in this chapter. As observed in both plots, GarNet becomes confident over time, and the early-stop strategy activates if 80% of decision points in the garment similarity map is within a correct category.

Chapter 5

Learning Physics Properties of Fabrics and Garments with a Physics Similarity Neural Network

Chapters 3 and 4 introduce predicting the shapes and visually perceived weights of garments. However, garments have other important properties for robotic garment manipulation, such as area weights and bending stiffness parameters. Some physics property parameters (such as bending stiffness) are difficult to measure in real environments directly [11]. However, these parameters are easily accessible in simulated environments. Suppose the learning and predicting of these physics property parameters of real fabrics and garments can be accomplished by learning from these objects in simulation environments. In that case, the problem of the difficulty in measuring these physics property parameters can be solved. However, garments consist of components such as collars, sleeves, buttons and pockets, which take a long time and are computationally costly to simulate. Therefore, this chapter investigates whether learning and predicting these physics property parameters of real garments can be achieved by learning the physics similarities between simulated fabrics, which are easier to be simulated. This chapter [1] *demonstrates that predicting the physics properties of complicated-structured objects such as garments can be achieved by learning the physics similarities between simply-structured objects such as fabrics.*

[1] *This chapter has appeared in [134]. Li Duan is the first author and main contributor to this paper.*

5.1 Introduction

This chapter proposes to predict the physics parameters of real fabrics and garments by learning the physics similarities between simulated fabrics via a Physics Similarity Network (PhySNet). For this, wind speeds generated by an electric fan and area weights were estimated to predict the bending stiffness of simulated and real fabrics and garments. This chapter found that PhySNet coupled with a Bayesian optimiser can predict physics parameters and improve the SOTA by 34.0% for real fabrics and 68.1% for real garments.

5.2 Motivation and Objectives

Robotic perception and manipulation of deformable objects are challenging due to the high dimensionality of their object states (i.e. complicated configurations), which are difficult to monitor and predict. Due to the object's complicated configurations and random deformations, a three-step process is usually adopted. The first step consists of modelling the objects in a simulated environment [135], and [136] or using finite element methods (FEM). Then, the second step is about learning deformations of the object in the simulated environment while the object is manipulated [80, 66]. The final step comprises finding an optimised trajectory for manipulating the object [73, 137]. In these three steps, the challenge is to learn the stress-strain curve of these deformable objects [138] which depends on the physics properties of objects such as stiffness, area weight and damping factors. Therefore, learning the physics properties of deformable objects is key to enabling a robot to perform dexterous manipulation of deformable objects.

As discussed in Chapter 2, previous approaches that estimate the physics properties of materials consist of either learning physics properties by aligning simulated models with real objects ([139], [34] and [30]), or learning physics properties from video frames ([11], [5] and [1]). The former approaches require high accuracy in aligning objects using the finite element methods (FEM), which are computationally costly. In contrast, the latter approaches do not need simulated models to match real objects. Therefore, learning from video sequences is computationally efficient and can be deployed in a robotic system where a robot can apply the external force on deformable objects. Simulating garments is time-consuming and computationally costly because they contain various components: packers, collars, sleeves and buttons. However, simulating fabrics is easy and achievable in simulation engines (Blender [135] and ArcSim [140]). If learning and predicting the physics properties of real garments can be achieved by learning the physics similarities of simulated fabrics, the above problems (difficult to be measured in real environments and time-consuming and computationally costly to be simulated) can be solved.

For robotic deformable object manipulations, the physics properties of deformable objects are linked with the deformation patterns of manipulated garments. For example, stiffer garments tend to bend less than softer garments, and softer garments have more complex states than rigid garments. It is assumed that if a robot has prior knowledge of the physics properties of garments, it can use these parameters to fine-tune a manipulation plan, making the garments' manipulation effective. This chapter proposes to learn the physics similarities between simulated fabrics to predict the physics properties of real fabrics and garments. For this, a Physics-Similarity Neural Network (PhySNet; inspired by [1]) has been implemented, as shown in Fig. 5.2, with the aim of predicting real fabric physics parameters from simulated fabric physics parameters. The core idea here is that measurements of physics properties are difficult to obtain in a real environment setting without the need of specialised equipment. For example, Bouman *et al.* [11] obtained experimentally fabric's stiffness parameters using specialised devices and designed a neural network architecture to regress stiffness parameters. Therefore, taking advantage of simulation software, where the physics properties can be obtained, can avoid the challenge of obtaining them in real environments. Hence, *is it possible to leverage a simulation engine to estimate physics parameters with a neural network when a wind force field is applied to the fabrics and garments*. To answer these questions, a simulated fabric database was compiled to allow PhySNet to learn the physics similarity of simulated fabrics and to generate a Physics Similarity Map (PSM) for a fabric. After a PhySNet was trained, a piece of real fabric or garment and a simulated fabric with initialised physics parameters are input into PhySNet to get their a Physics Similarity Distance (PSD). The PSD is then input into a Bayesian optimiser, which outputs updated physical parameters. The updated physics parameters are input into the simulator to generate a new simulated fabric. This procedure iterates until stable parameters (Section 5.5.2) are obtained from the Bayesian optimiser. PhySNet is trained on simulated data which is used to learn physical similarities between simulated fabrics. During testing, real garments are input into the PhySNet, and simulated fabrics are matched with optimal physics distances. The novelty of this chapter is that *the experiments test on the real garments that do not appear in the training database, indicating that the PhySNet is tested on unseen garments.* The contributions of this chapter are threefold:

1. This chapter proposes that physics property parameters of real fabrics and garments can be predicted from learning physics similarities between simulated fabrics;
2. This chapter proposes that predicting the physics property parameters of real complicated objects such as garments can be achieved from learning physics similarities between simulated simple objects such as fabrics;

3. This chapter demonstrates that learning physics property from depth images outperforms learning them from RGB images.

5.3 Fabric Physics Properties

The relationship in bending stiffness between strain and stress, as given by [30], is:

$$F = k_e sin(\theta/2)(h_1 + h_2)^{-1}|E|\,u \tag{5.1}$$

where F is the external force, and k_e is the material's bending stiffness. Figure 5.1 shows a visualisation of Eq. 5.1. In Figure 5.1, triangles 123 and 143 represent two faces of a piece of fabric where a force is applied to bend the fabric from triangle 123 to triangle 143. h_1 and h_2 are the normals of the two triangles, while E is an edge vector of the edge 13 which is shared by both the triangles 123 and 143. u is a bending model described in [30]. In Eq. 5.1, Wang et al. [30] treated the bending stiffness, k_e, as a linear piecewise function of the reparametrisation $sin(\theta/2)(h_1 + h_2)^{-1}$. To estimate bending stiffness, the *bending angle*, θ (in Fig. 5.1), are set to 0°, 45° and 90°. The bending stiffness is measured five times for each value of θ. These five measurement points represent the bending behaviours of a piece of fabric in [30] experiments. Therefore, there are 15 points represented by a matrix of size 3×5 (angles $\times$ bending measurement points). Predicted bending stiffness of the fabrics is represented using this matrix representation (e.g. Figure 5.6).

Bending stiffness is difficult to be measured directly without specialised devices [11], but bending stiffness can be derived from the strain-stress curve of materials [138]. Therefore, if a neural network can learn the strain-stress relationship, it is possible to estimate the bending stiffness of fabrics and garments. By observing deformations of fabrics and garments, if the predicted external forces (stresses) match measured external forces and deformations between simulated and real fabrics and garments, it can be established that the predicted bending stiffness can be approximated to the real values. The match between deformations of real and simulated objects is referred to as *Physics Similarity Distances* (PSD, Section 5.4.1).

In the experiments, an electric fan was used to wave real fabrics to exert an external force. Therefore, wind speed is predicted, which is proportional to wind force, as $F_w = 1/2A\rho v$ where F_w is the wind force, ρ is the air density, A is the surface area of a deformable object and v is the wind speed. In the experiments, the fabrics used in the experiments have a surface area of 1 m^2.

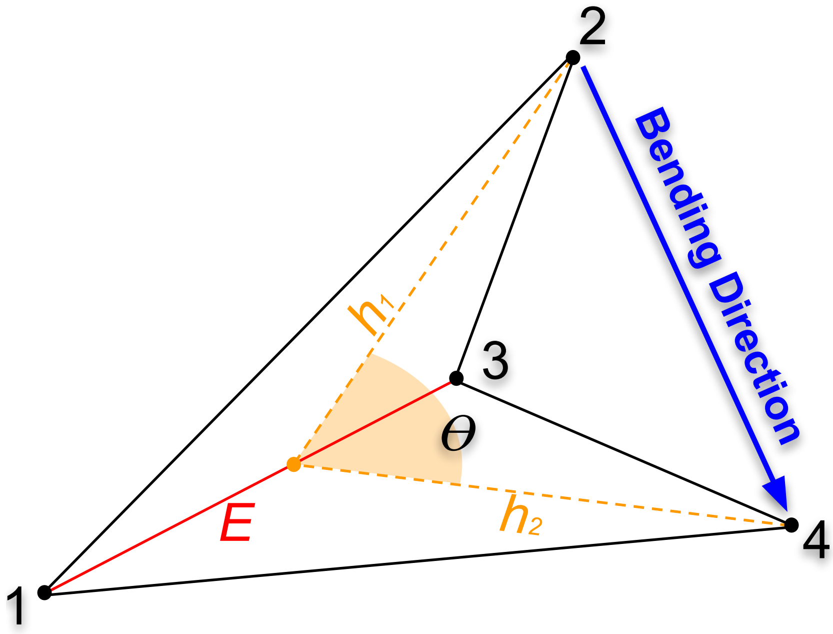


Figure 5.1: Bending Stiffness: h_1 and h_2 are normals of the triangles 123 and 143, E is the edge vector of the edge 13.

5.4 Materials and Methods

5.4.1 PhySNet

In this chapter, a Physics Similarity Network (PhySNet) was proposed, which is a Siamese network [141, 1] that clusters input data according to their labels. PhySNet comprises a convolutional neural network that extracts features from input data and a fully connected layer that maps the extracted features into a 2D *Physics Similarity Map* (PSM). The PhySNet is expressed as$P = f_\theta(I)$, where f_θ denotes a neural network that contains convolutional layers and fully connected layers parameterised by the parameters θ, and I denotes an input video frame. P is defined as a *physics similarity point*, which is a point on the PSM mapped from an input fabric image I. With these points in the PSM, a *Physics Similarity Distance* (PSD) is defined as:

$$PSD_{i,j} = \left|P_i - P_j\right|^2 \tag{5.2}$$

where i and j are the *i-th* and *j-th* physics similarity points in the PSM of two different fabric images. Input fabric images can either be RGB or depth images of fabrics labelled according to their physics properties and external parameters.

As opposed to [1] where they have used a contrastive loss on pairs of positive and negative samples, this chapter proposes to use a triplet loss function based on observations in [142].

Images are triplet-classed, meaning that every input contains three images, one defined as an anchor and the other as positive and negative samples of the anchor. The input triplets are mapped onto the PSM through PhySNet as physics similarity points. Thus, the loss function is defined as:

$$\begin{aligned} PP &= \left|P_{positive} - P_{anchor}\right|^2 \\ NP &= \left|P_{negative} - P_{anchor}\right|^2 \\ Loss &= max(0, PP - NP + Margin) \end{aligned} \tag{5.3}$$

where $P_{positive}$, $P_{negative}$ and P_{anchor} are the positive, negative and anchor points, respectively. An anchor point is a point output from the PhySNet with an input of an image of a piece of fabric. A positive point is a point output from the PhySNet with an input of an image of a piece of fabric of the same physics property parameters as the anchor one. A negative point is a point output from the PhySNet with an input of an image of a piece of fabric with different physics property parameters to the anchor one. PP and NP are the positive pair and negative pair, respectively. A positive pair is a pair of points output from PhySNet with the inputs of the images of two fabrics with the same physics property parameters (bending stiffness parameters and area weights). In comparison, a negative pair is a pair of points output from PhySNet with the inputs of the images of two fabrics with different physics property parameters. The loss function aims to shorten the Physics Similarity Distances (PSDs) between the positive pairs and increase the PSDs between the negative pairs. A margin is a value which ensures that large PSDs do not contribute to the gradient updates of the network thus that the network concentrates on the pairs that have small PSDs. The margin in the experiments is set to 1 (as described in [1]).

5.4.2 Bayesian Optimisation

Initialised physics parameters are input into the simulation engine and output simulated fabrics. The simulated fabrics and real fabrics are fed into a trained PhySNet, which outputs their physics distances. This chapter aims to close the gap between simulation and reality by finding the simulation's physics parameters that resemble those observed in reality. For this, a Bayesian optimisation was used to find these physics parameters for the simulation and the objective is to minimise physics distances between simulated and real fabrics.

To minimise physics distances, physics distances were thus converted into negative values (for example, convert a physics distance of 100 to -100) and maximise negative values (optimal values are 0). Botorch [143] has been used to implement Bayesian Optimisation. Figure 5.2 shows the proposed pipeline.

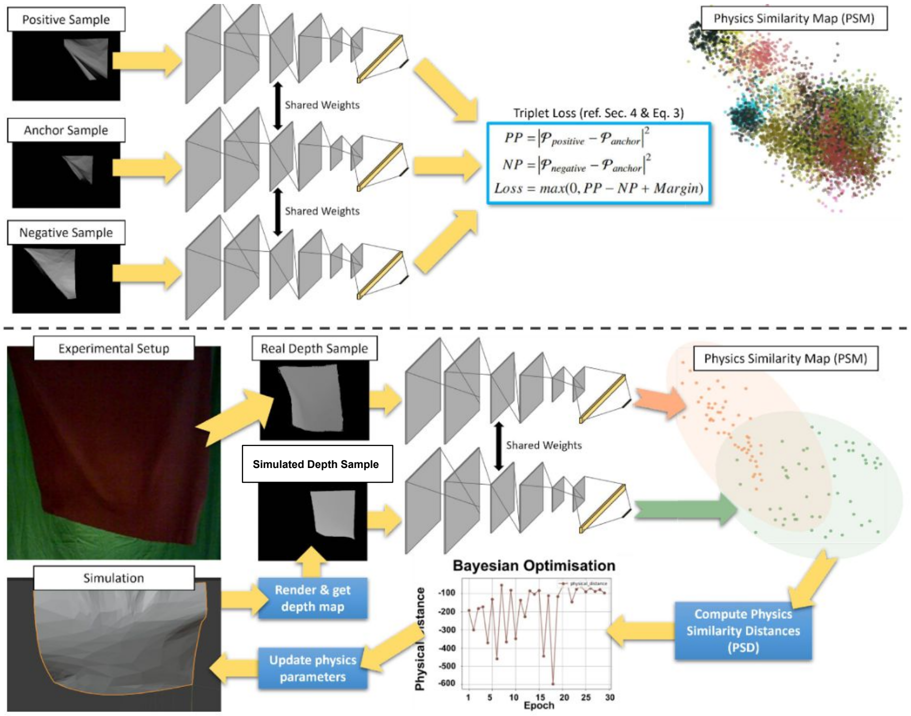

Positive Sample
Anchor Sample
Negative Sample
Shared Weights
Shared Weights
Triplet Loss (ref. Sec. 4 & Eq. 3)
$PP = |\mathcal{P}_{positive} - \mathcal{P}_{anchor}|^2$
$NP = |\mathcal{P}_{negative} - \mathcal{P}_{anchor}|^2$
$Loss = max(0, PP - NP + Margin)$
Physics Similarity Map (PSM)
Experimental Setup
Real Depth Sample
Simulated Depth Sample
Shared Weights
Physics Similarity Map (PSM)
Simulation
Render & get depth map
Update physics parameters
Bayesian Optimisation
physical_distance
Physical Distance
-100
-200
-300
-400
-500
-600
1
5
10
15
20
25
30
Epoch
Compute Physics Similarity Distances (PSD)

Figure 5.2: (*Top*): A triplet of images (an anchor, a positive and a negative) are input into PhysSNet, and a Triplet loss function (Eq. 5.3) is used to learn whether fabrics in the images have the same or different physics properties. After training, PhysSNet generates a Physics Similarity Map (PSM), where fabrics with similar physics properties have smaller Physics Similarity Distances. (*Bottom*): An Xtion camera were used to capture real fabrics and an electric fan were used to exert a force onto the fabrics. The depth images of a simulated fabric with initial physics property parameters and the depth images of a real fabric are input into the trained PhysSNet to have them mapped on the physics similarity map. The output of PhysSNet is physics similarity points mapped on the physics similarity map, as shown in the training and testing parts in the figure. Their PSDs are calculated from the map and are input into a Bayesian optimiser. The Bayesian optimiser outputs optimal physics property parameters which are used to generate a new simulated fabric. This loop iterates until the differences between optimal physics property parameters of the last three iterations are less than 10%

5.5 Experiments

5.5.1 Fabrics and Garments Dataset

For the experiments, both simulated and real fabric samples were collected. To simulate fabrics, ArcSim [140] was used which is a deformable object simulator that uses triangle meshes and linear piecewise functions (Section 5.3). Inputs to ArcSim are the physics parameters of fabrics, including stretching stiffness, bending stiffness and area weights, and external environmental parameters, including gravity, wind speed and wind direction. In this experiment, the search space for the Bayesian Optimisation (as defined in Section 5.4.2) includes bending stiffness,

Table 5.1: Area Weight Search Space for Different Materials

Material	Area Weight Search Space
White Tablecloth	0.1-0.17 m/s^2
Gray Interlock	0.15-0.22 m/s^2
Black Denim	0.30-0.37 m/s^2
Sparkle Fleece	0.23-0.30 m/s^2
Pink Nylon	0.16-0.23 m/s^2
Ponte Roma	0.23-0.3 m/s^2
Red Violet	0.1-0.17 m/s^2

wind speed and area weight; thus, other parameter settings were kept in their default values. The external parameters are: (i) wind speed (from 1 to 6 m/s), (ii) fabric's area weight (see Table 5.1), and (iii) Bending stiffness (from 0.1 to 10 times of standard bending stiffness parameters, ref. [30, 1]). This search space was defined based on the experimental settings described in [1].

Seven different materials have been tested ; *tablecloth*, *interlock*, *denim*, *sparkle fleece*, *nylon*, *ponte roma* and *jet set (red-violet)*. These materials were chosen because they are common in the textile industry. Table 5.1 shows the search space for different materials in terms of their area weights, which are obtained from the manufacturers. The search space was set for wind speeds to 1-6 m/s.

ArcSim outputs a sequence of 60 3D models. The length of each video is 3 seconds with a sampling frequency of 20Hz. These 60 3D models were input into Blender [135] to render them into a video sequence of depth images, where each 3D model corresponds to one frame. Because depth images are sensitive to cameras' relative positions with respect to the captured object, randomising cameras' positions in the simulation environment can enhance PhySNet to recognise real fabrics and garments. Therefore, the locations of the camera were randomised in Blender and captured a fabric from six different locations. That is, this chapter translates the x (from 1 to 6) and z (from -0.5 to 0.3) axes in ArcSim while leaving fixed the y axis to 0.5. Similarly, the camera was rotated in ArcSim for z (from -260° to 280°), x to 90° and y to 0°. Bending stiffness settings are referenced in [30], where they provide measured values of the materials that were used in the experiments. Therefore, the search space for bending stiffness is from 0.1 to 10 of measured values in [30].

For each simulated material, 30 different combinations of physics properties and external environmental parameters were randomised and constrained within the search space defined above. Combinations are uniformly distributed, and each combination comprises a sequence of 60 3D models. These 60 models were input into the Blender engine, which renders the models with 6 rendering camera positions. Therefore, 10,800 images for each material were captured, which are labelled with their combination number.

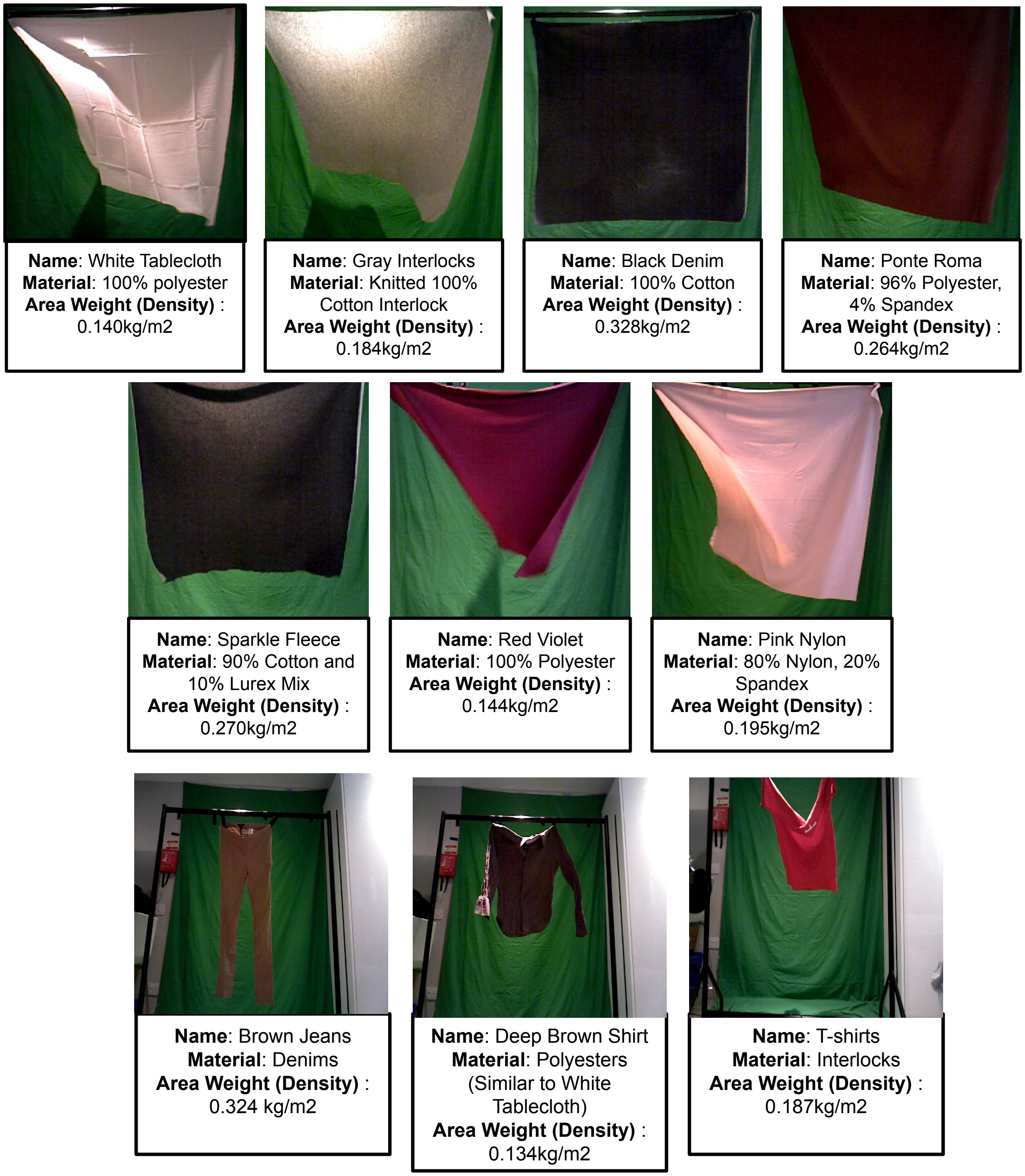
An Asus Xtion camera was used to collect real fabric and garment samples. An electric fan waves fabrics with wind speeds varying from 2.4-3.1 m/s. The varying wind speeds can test whether the approach can detect fabrics and garments' physics properties under different wind speeds. For each real sample, a video of 60 frames in length is recorded at a sampling frequency of 24 fps (2.5 s in real-time for each video). Wind speeds are measured by an electronic anemometer (model AOPUTTRIVER AP-816B) and area weights are measured using an electric scale. All fabrics are cut into a square of 1 m × 1 m such that their weights scaled by the electric scale are unit area weights. The testing points for wind speeds are located near the fabric.The experiment environment has a normal level of humidity (2.8 g/m3 absolute humidity or 30% relative humidity) and temperature (10 degrees). These conditions remained constant as the experiments were conducted in an indoor environment where environmental conditions were kept constant. The experiments and results were not affected by the humidity and temperature conditions. Figure 5.3 shows the dataset for real garments and fabrics used in the experiments.

5.5.2 Experimental Methodology

PhySNet has been implemented in Pytorch. PhySNet consists of 2D convolutional layers with a PReLU layer and a MaxPool2D layer between adjacent convolutional layers. The convolutional layers are followed by a fully connected layer with three linear layers and a PReLU between adjacent linear layers. Input images are 1-channel depth with an image resolution of 256×256. an Adam optimiser has been used with a batch size of 32 and a learning rate of 10^{-2}. A learning scheduler with a step size of 8 and a decay factor of 10^{-1} has been used for the optimiser. The PhySNet was trained for 30 epochs with a batch size of 32. The PhySNet is trained on simulated fabrics images but tested on real, unseen fabrics and garments. Figure 5.4 demonstrates the details of the proposed network.

The range of wind speed is from 2.5m/s to 3.5m/s depending on the electrical fan settings. The experiment tests fabrics and garments under different wind speeds because wind speeds are variant in the real world. This chapter uses wind speeds as an indicator of bending stiffness parameters (which means bending stiffness parameters are an approximation) because bending stiffness parameters are difficult to measure without specific instruments ([11]). The bending stiffness parameters are intrinsic properties of fabrics and garments, which means the position of the fabrics and garments will not change their bending stiffness parameters.

The performance of PhySNet was compared with the Spectrum Decomposition Network (SDN) proposed in [1]. The SDN is a network that uses a Fourier transformation to convert time-domain RGB images into frequency-domain maps and extracts top K maximum-frequency parts of the



Name: White Tablecloth
Material: 100% polyester
Area Weight (Density) : 0.140kg/m2

Name: Gray Interlocks
Material: Knitted 100% Cotton Interlock
Area Weight (Density) : 0.184kg/m2

Name: Black Denim
Material: 100% Cotton
Area Weight (Density) : 0.328kg/m2

Name: Ponte Roma
Material: 96% Polyester, 4% Spandex
Area Weight (Density) : 0.264kg/m2

Name: Sparkle Fleece
Material: 90% Cotton and 10% Lurex Mix
Area Weight (Density) : 0.270kg/m2

Name: Red Violet
Material: 100% Polyester
Area Weight (Density) : 0.144kg/m2

Name: Pink Nylon
Material: 80% Nylon, 20% Spandex
Area Weight (Density) : 0.195kg/m2

Name: Brown Jeans
Material: Denims
Area Weight (Density) : 0.324 kg/m2

Name: Deep Brown Shirt
Material: Polyesters (Similar to White Tablecloth)
Area Weight (Density) : 0.134kg/m2

Name: T-shirts
Material: Interlocks
Area Weight (Density) : 0.187kg/m2

Figure 5.3: *Fabric and Garment*

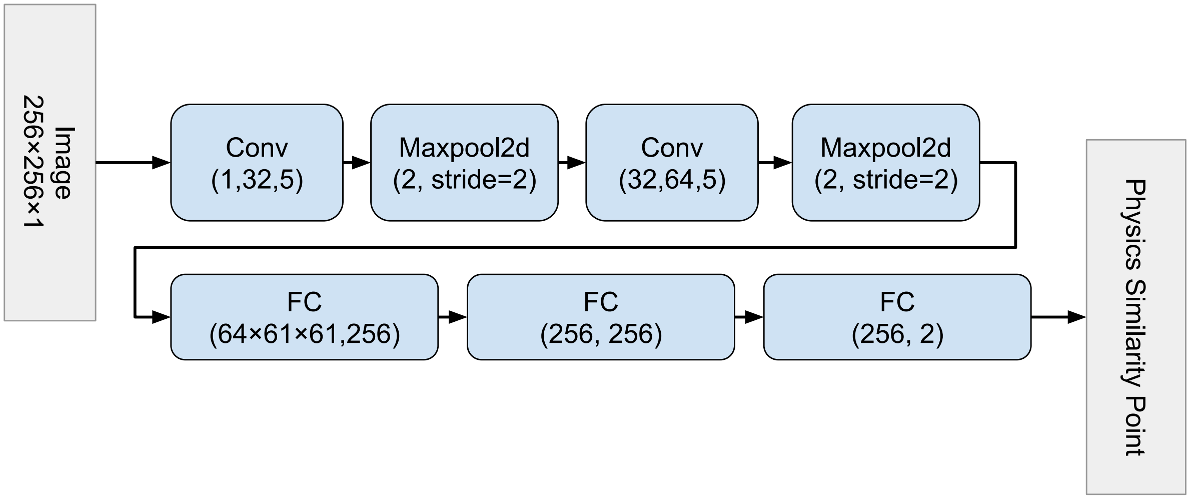


Figure 5.4: *The details of the proposed network*

maps as features. For baselines, the performance of four networks was compared: two networks are PhySNet trained on depth and RGB images, and the other two are SDN trained on depth and RGB images.

Estimating Physics Parameters of Fabrics and Garments

This experiment aims to find the physics and external environmental parameters of real fabrics. Therefore, parameter settings were adjusted in the simulation engine to generate a simulated fabric and calculate its PSD to the real fabric on the PSM. The optimization were halted once a stable PSD was found between a simulated and real fabric or garment (ref. Section 5.4.2). As discussed in Section 5.3, only predicted results of wind speeds and area weights were compared because there was no ground truth for the bending stiffness of the real fabrics, but wind speeds serve as indicators of bending stiffness of the real fabrics and act as the ground truth to validate the proposed approach.

The Bayesian Optimiser described in section 5.4.2 is used to find physics and external environmental parameters for simulated fabrics that can minimise the physic similarity distance between simulated fabrics and real fabrics or garments. Parameters optimised in this experiment are bending stiffness, wind speeds and area weights, which are normalised to $[-1,1]$. Values for the parameters are initially set as 0. The search space for these parameters is the same as the search space set for simulated data as in Section 5.5.1. The Bayesian Optimisation was halted when updated parameters became stable. Parameter updates do not change by more than 10% over the last three epochs. Wind speed and area weight estimations are compared with the measured ground truths, i.e. from the anemometer and electric scale.

Table 5.2: Clustering Accuracy for PhySNet and the SDN networks [1] trained on depth and RGB images.

Name	PhySNet (Depth)	SDN (RGB)	PhySNet (RGB)	**SDN (Depth)**
White Tablecloth	80%	**91**%	89%	90%
Black Denim	88%	89%	86%	**97%**
Gray Interlock	83%	86%	84%	**92%**
Sparkle Fleece	77%	82%	78%	**92%**
Ponte Roma	79%	84%	80%	**93%**
Pink Nylon	80%	83%	77%	**93%**
Red Violet	80%	87%	78%	**92%**

Simulating fabrics is easier than simulating garments because fabrics have simple geometric shapes whereas garments have complex shapes. If PhySNet can recognise real garments while being trained on simulated fabrics, simulating complicatedly-structured garments can therefore be bypassed. Three garments were selected: a T-shirt, a shirt and jeans. To measure the physics parameters of these garments, PhySNets trained on the grey interlock (for the T-shirt), a white tablecloth (for the shirt) and black denim (for the jeans) were used because these garments are made of these fabrics and have similar physics parameters.

The electric fan waves garments and the wind speeds are recorded using the anemometer. Likewise, the same methodology was followed for fabrics to capture garments as video sequences. Garment images are input directly into PhySNet, and the Bayesian optimiser is used to find the garments' physics parameters. A garment is firstly compared with a simulated fabric of the same material rendered with parameters set to 0. Updated parameters from the Bayesian optimiser are input into the simulator to output an updated fabric, and it is then compared to the real garments until stable parameters are obtained. the Bayesian Optimisation was halted when updated parameters became stable as in the fabrics experiment.

5.6 Experimental Results

5.6.1 Clustering Accuracy of PhySNet and SDN

Table 5.2 shows that the best performance for clustering accuracy is on the SDN-trained network while using depth images. Whereas the network with the lowest accuracy is PhySNet trained on depth images. Overall, SDN has a better performance than PhySNet. This is because a Fourier transform outputs a frequency map for the transformed images, and on this frequency map, areas of the fabrics that deform fast from the waving wind are amplified while static areas are attenuated. The SDN benefits from these frequency maps while ignoring ‘less deformed’ areas, but this causes an information loss and overfitting of the training data. This loss of information can potentially reduce the network’s ability to recognise real fabrics as shown in Section 5.6.2.

From Table 5.2, PhySNet trained on RGB images has a better performance than PhySNet trained on depth images. For depth images, changes in physics parameters do not have the same levels of influence on spatial characteristics as texture characteristics. Depth information remains relatively constant between simulated and real fabrics, and this means that depth is suitable for finding the physics parameters of real fabrics and generalising better across domains.

5.6.2 Predicting Fabrics’ and Garments’ Physics Parameters

It can be observed in Table 5.3 that the best performance is obtained using the PhySNet trained on depth images. The approach improves the SOTA (SDN trained on RGB images) by 34.0%. Both the SDNs (trained on depth and RGB images) experience failures in finding the physics parameters of real fabrics (denoted as ’F’). The reason for the failures is that the SDN failed to correctly map real fabric or garment images onto the physics similarity map; hence, the Bayesian optimiser cannot find optimal values for the physics parameters of real fabrics. As discussed in Section 5.6, the SDN has the disadvantage of information loss that affects the network’s ability to predict the physics properties of real fabrics and garments. From Table 5.3, It also can be observed that PhySNet trained on depth images outperforms PhySNet trained on the RGB images. Depth images directly capture deformations while RGB images capture changes in the texture and colour manifolds that are not descriptive of deformations and structural changes.

The proposed PhySNet is better than SOTA for two reasons. Firstly, PhySNet outperforms SOTA by 34.0% on fabric prediction and 68.1% on garment prediction. Secondly, failure cases exist in SOTA, while PhySNet succeeds in predicting all fabrics and garments. Only fabrics and garments for which PhySNet and SOTA succeed in prediction are compared with their accuracies.

Figure 5.6 shows the predicted bending stiffness of real fabrics. Bending stiffness parameters are represented as matrices (as defined in Section 5.3). Therefore, surface plots were used to display predicted values. From Figure 5.6, it can be observed that *black denim* is the stiffest material, while the *sparkle fleece* is the softest material because *black denim* has the highest predicted bending stiffness while *sparkle fleece* has the lowest predicted value. These measurement results align with human intuitions, where denim (i.e. jeans) is stiffer than sparkle fleece (i.e. sweaters).

Table 5.3: *Fabric Physics Parameter Estimation.* Percentage errors are w.r.t group truths; wind (unit: m/s) and area weight (unit: kg/m^2).

Materials	PhySNet (depth)	PhySNet (RGB)	SDN (depth)	SDN (RGB)
White Tablecloth	**6.5%, 8.6%**	9.2%, 10%	119.6%, 15.0%	75.4%, 11.11%
Gray Interlock	9.6%, 19.6%	40.7%, 10.9%	5.4%, 15.8%	**5.4%, 3.3%**
Black Denim	**5.4%, 5.5%**	37.3%, 1.2%	F, F	90.0%, 8.2%
Ponte Roma	35%, 0.4%	34.6%, 0.4%	**22.7%, 0.8%**	33.8%, 0%
Sparkle Fleece	**34.6%, 0%**	**34.6%, 0%**	48.8%, 2.6%	F, F
Red Violet	**16.7%, 6.3%**	**16.7%, 6.3%**	F, F	F, F
Pink Nylon	**12.6%, 2.6%**	57.1%, 2.6%	F, F	F, F

Table 5.4: *Garment Physics Parameter Estimation.* Percentage errors are w.r.t group truths; wind (unit: m/s) and area weight (unit: kg/m^2).

Materials	**PhySNet (depth)**	PhySNet (RGB)	SDN (depth)	SDN (RGB)
T-shirt	3.1%, 17.1%	**1.92%, 15.5%8**	F, F	27.3%, 0.5%
Deep Brown Shirt	**34.2%, 1.5%**	59.2%, 6.7%	60.4%, 18.7%	F, F
Brown Jeans	**21.7%, 0.6%**	97.9%, 2.5%	F, F	148.3%, 8.3%

Table 5.4 shows the Bayesian Optimisation results for garments. It can be observed that PhySNet trained on depth images has the best performance when predicting garments' physics properties and external environmental parameters. However, Table 5.4 shows that predictions for garments are not as accurate as the predictions for fabrics due to the different shapes between the garments and fabrics. SDN RGB and depth and the PhySNet RGB failed to optimise correctly and converged to incorrect values for each of the three garments. The results, similar to section 5.6.2, indicate the disadvantages of using RGB images and frequency maps for finding real-garment physics parameters. Predicted stiffness parameters are shown in Figure 5.6. It can be observed that jeans are stiffer than T-shirts and shirts, which aligns with human intuition. These results suggest that it is possible to estimate the physics properties of garments by training PhySNet on simple fabrics with a mean average error of 17.2% for wind speeds and 6.5% for area weight parameters. Overall, the obtained performance improvement between the proposed approach (PhySNet on depth images) and SDN on RGB images (state of art) is 68.1%

5.7 Conclusion

This chapter proposed that predicting the physics property parameters of real fabrics and garments can be achieved by learning the similarities between simulated fabrics. PhySNet outperforms SOTA (SDN) by 34.0% for fabrics and 68.1% for garments. This chapter reveals that

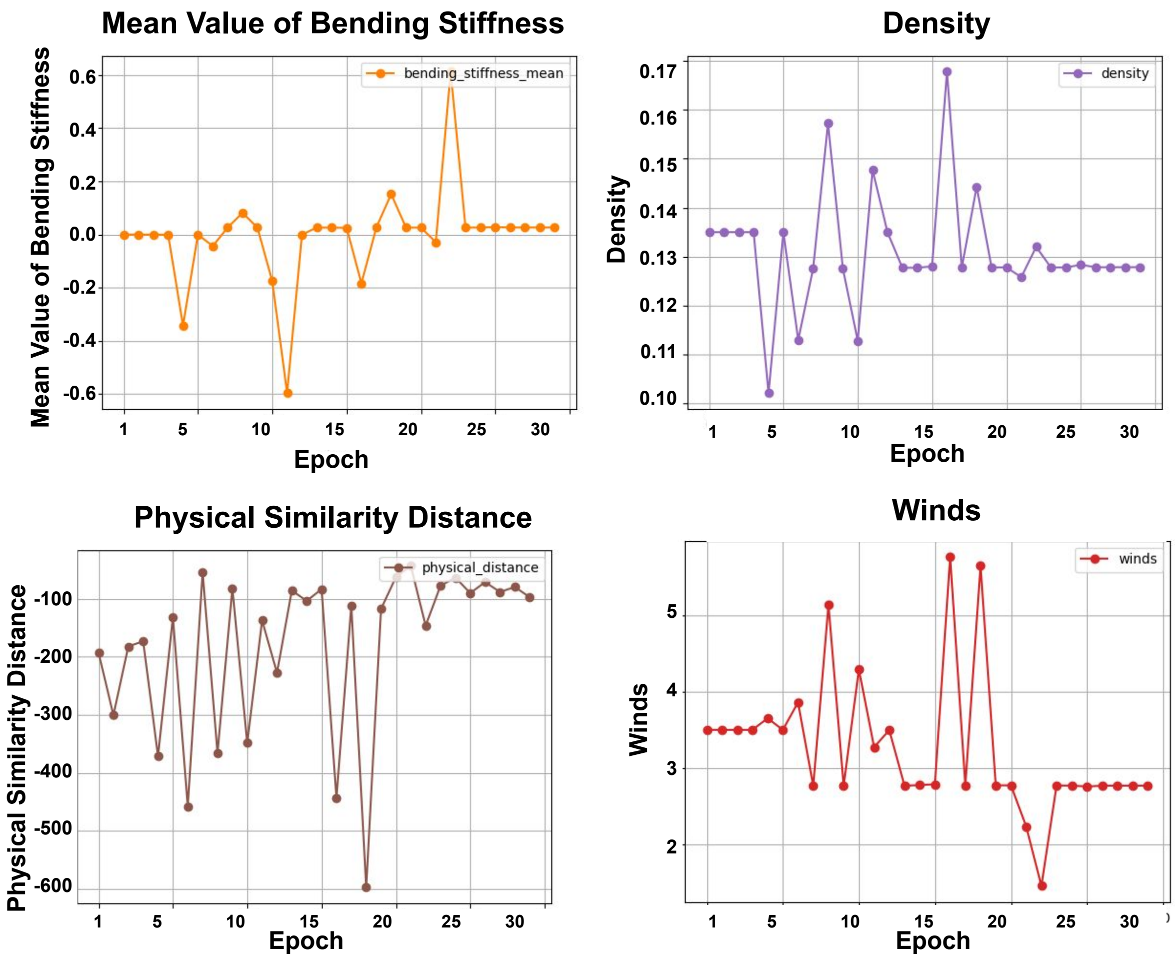


Figure 5.5: An example of a successful Bayesian Optimisation: PhySNet estimating the physics parameters of the white tablecloth.

learning the physics properties of real and complicatedly-structured objects (such as garments) can be achieved via learning the physics similarities between simulated and simply-structured objects (such as fabrics), which addressed the limitations discussed in Section 2.2 of Chapter 2. However, there are limitations to the proposed approach. That is, only bending stiffness is considered, and physical properties that determine strains (deformations) consist of stretching stiffness, bending stiffness and damping. The reason to limit the physics parameters is to reduce the search space for the Bayesian Optimisation and guarantee convergence.

Further research consists of developing a better optimisation method to optimise all physics properties. PhySNet is proven more effective while being trained on one rather than multiple materials. Future research focuses on devising a methodology to enable a neural network to be trained on different materials and predict the physics properties of different fabric materials. Indeed, a general purpose of using PhySNet for predicting the physics property parameters of fabrics and garments is envisaged to facilitate robotic fabric and garment manipulation.

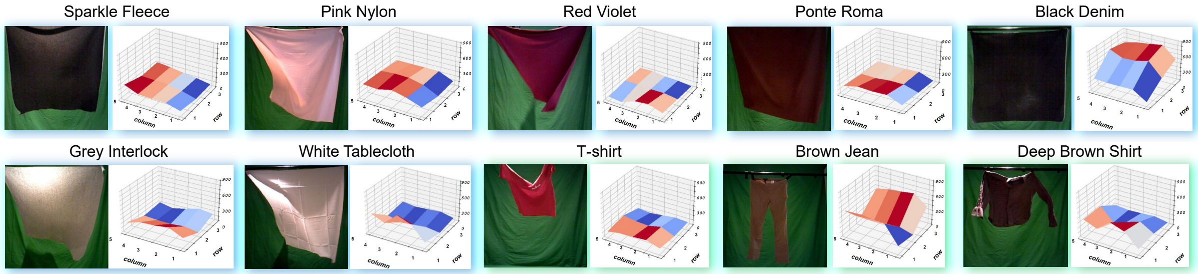


Figure 5.6: *Predicted Bending Stiffness of Real Fabrics and Garments.* surface plots were used to visualise predicted bending stiffness parameters of real fabrics and garments. The x and y axes in the surface plots represent the row and column indexes of the parameters, and the z axes are the bending stiffness parameters.

In the experiments, an electric fan was used to exert an external force (waving) on fabrics and garments. a robot was also proved to be able to interact with garments [125], [3], and robots are envisaged to be able to exert these forces on fabrics and garments while the robot interacts with the objects. That is, a robot can stretch objects to measure stretching stiffness and facilitate manipulating objects by grasping and dropping them to observe their deformations. From these interactions, the network can be effective in learning the physics parameters of deformable objects.

This chapter verified the following hypothesis: Predicting the bending stiffness parameters and area weights of real garments and fabrics can be achieved by learning physics similarities between simulated fabrics with a Siamese network, which outperforms SOTA by at least 30%;

Chapter 3, Chapter 4 and Chapter 5 introduce three robotic perception paradigms for learning garment attributes (shapes, visually perceived weights and physics properties). These attributes are essential for robotic garment manipulation. Chapter 7 shows an effective robotic garment flattening pipeline that incorporates garment shape information to improve manipulation efficiency, demonstrating the necessity of learning garment attributes in robotic garment manipulation. However, this thesis does not incorporate garment physics property information in the proposed pipeline, but a discussion on how to incorporate garment physics properties in the proposed pipeline (also in other robotic garment manipulations) in future work is presented in Chapter 8.

Chapter 6

Recognising Known Configurations of Garments For Dual-Arm Robotic Flattening

Robotic garment manipulation is the second part of this PhD thesis, where garments' geometric properties (shapes) are utilised to improve manipulation performances. Chapter 6 introduces robotic garment flattening based on recognising the known configurations of garments. Known configurations are garment configurations hanging in the middle of the air after a robot grasps them up. This chapter [1] *demonstrates that garments' complicated starting configurations can be converted into simple and unified known configurations. Pre-designed manipulation strategies can be utilised to manipulate garments rather than real-time updating manipulation strategies, effectively reducing manipulation times. Chapter 7 introduces improving the recognition of known configurations for garments by utilising the garment similarity network mentioned in Chapter 4 to predict garment shapes. Combining chapters 6 and 7 provides a full pipeline of effective robotic garment manipulation.*

6.1 Introduction

Predicting deformable object states and updating manipulation strategies is time-consuming and computationally expensive. This chapter proposes learning *known configurations* of garments to allow a robot to recognise garment states and choose pre-designed manipulation strategies to

[1] *This chapter has appeared in [20]. Li Duan is the first author and main contributor to this paper.*

flatten these garments.

6.2 Motivation and Objectives

Robots have been widely applied in many aspects of life, such as robotic assistance for humans, life-threatening rescue, and manufacturing. Among these applications, robotic deformable object manipulation plays a critical part. Autonomous garment laundry and sorting, soft-object manipulation in manufacturing, and robotic weaving in the textile industry require an efficient and robust robotic deformable-object manipulation strategy.

However, robotic manipulation of deformable objects remains an open problem in robotic research because deformable objects can take unpredictable object states (crumpled, stretched, or bent configurations). That is, their states frequently vary when they are being manipulated and cannot be predicted precisely. Deformable objects are easily slipped from robots and take a long time to be manipulated because of their properties. Therefore, SOTA focuses on reducing object-state dimensionality or robotic action-state complexity.

As discussed in Chapter 2, SOTA approaches that reduce deformable object complexity for robotic manipulation can be divided into two categories: these that propose to find simplified models to represent object states or action states [8, 144, 145, 73, 7, 6], and these that rely on model-free reinforcement or imitation learning solutions where the agent learns to model the object while interacting with it [84, 146, 9]. When deformable object models are simplified, it is possible to reduce the computational costs, making it possible to predict the object states of manipulated deformable objects close to real-time. An efficient prediction of future object states enables a robot to update its manipulation strategy on-the-fly, which is critical to manipulating deformable objects in a real-time setting. However, current research is geared towards simple geometries such as ropes, towels or cube-shaped objects. When a reinforcement learning solution is used, an agent learns manipulation policies using a reward-based mechanism. Robots are rewarded if their manipulation policies produce actions that contribute to a step closer to the object's goal state. However, most of the reinforcement learning approaches rely on simulated environments, which have a gap from real environments.

This chapter proposes to take advantage of gravity to model and learn the *known configurations* of deformable objects (garments in this chapter), which can be later used to choose a pre-designed manipulation strategy based on the recognised *known configurations* to flatten these objects (garments). The proposed approach does not need to find simplified models that provide good representations of object states for deformable objects. Similarly, the proposed approach does not require manual action labelling compared to reinforcement-learning approaches. This chapter focuses on recognising the *known configurations* of garments and developing a pipeline to flatten garments; see Figure 6.2. This chapter proposes a *known configuration* based robotic garment flattening pipeline, introducing a *known configuration* network (KCNet) and pre-designed manipulation strategies. This chapter addresses the challenges in sections 2.3.1, 2.3.2 and 2.3.3 in Chapter 2.

In summary, the main contributions in this chapter are two-fold:

- Learning the *known configurations* of garments is proposed, which is the foundation for a full pipeline of an effective robotic garment flattening pipeline in Chapter 7. Compared with SOTA, the proposed approach has a higher accuracy (89% versus 73% in [6, 7]);
- A *known configurations* database was captured which comprises RGB and depth images of garments.

In this chapter, the methodology is introduced in section 6.3, the experiment results are in section 6.4 and a discussion is in section 6.5.

6.3 Methodology

As discussed in Chapter 2, SOTA approaches conduct manipulations on tables or platforms where deformable objects have complex configurations (or object states) and result in computationally costly updates to find a manipulation strategy. In contrast, taking advantage of gravity to control the variety of object configurations and reduce the complexity of deformable object configurations has been considered.

This chapter assumes that garments of the same categories (e.g. jeans, towels, and t-shirts, etc.) lying on a table have different configurations. If the garments are grasped from similar grasping points, they will have similar configurations when the robot picks them up, which is called *known configurations* in this chapter. For example, in Figure 6.1 two towels have different crumpled configurations on a table (i.e. starting configurations), but they have similar *known configurations* after a robot grasps them. These *known configurations* only depend on the grasping points because of gravity. That is, complex starting configurations were converted to simple *known configurations* from which the robot can follow pre-defined manipulation strategies to

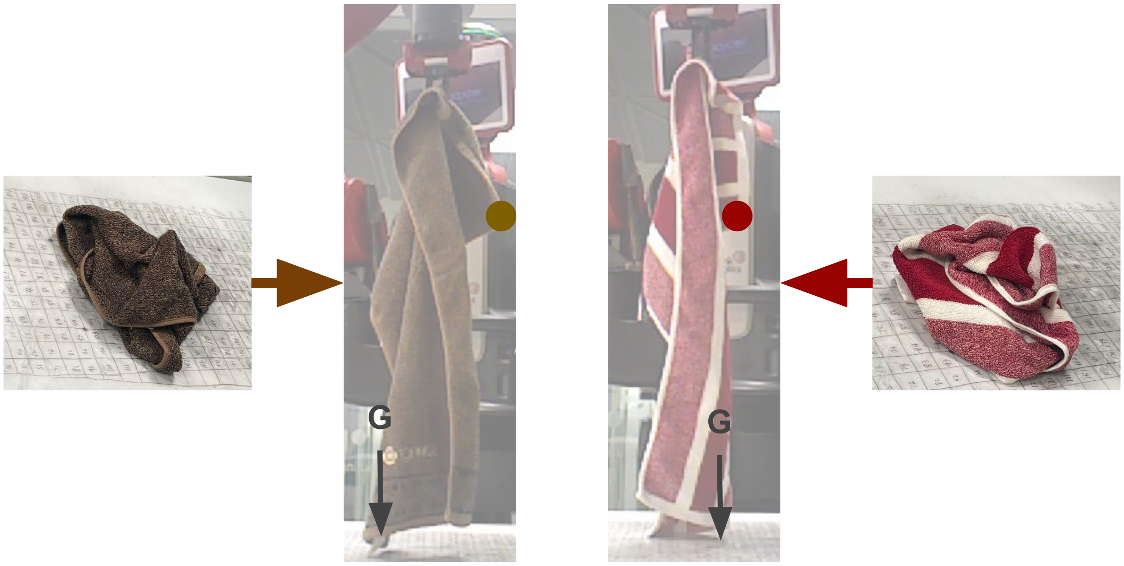


Figure 6.1: *Known Configurations principle*: Two towels have different initial configurations (starting configurations), but they form similar grasped configurations (*known configurations*) after they are grasped from similar grasping points.

flatten them. Recognising the *known configurations* of garments is crucial for the robotic garment flattening pipeline shown in Figure 6.2.

After the garment's *known configuration* is obtained, the second grasping point is found based on gravity's effect on the garment. For example, as shown in Figure 6.2 step 1 in the *manipulation strategy* box, the second grasping point is the lowest point of the towel. Different *known configurations* have different locations of the second grasping points, thus recognising the garment's *known configurations* is critical to localise the second grasping point. After the robot finds the second grasping point, the next step (step two in Figure 6.2) is to find the third grasping point, which is the opposing ending corner of the towel. Then, the robot stretches the towel from the two grasping points in step three in Figure 6.2. Because of the stretching and gravity, the towel is in a flattened state; therefore, the final step consists of placing the towel on the table by sliding it on the edge of the table (shown in steps four and five in Figure 6.2).

To recognise the garment's *known configurations*, a deep neural network (named KCNet in this chapter) was proposed based on ResNet-18[122]. KCNet is mathematically expressed as: $O = F(C(I))$, where O is the output of the KCNet, which is the recognised *known configurations*, I is the input image that captures *known configurations*, C are convolutional layers (a ResNet-18 [122]), and F are fully connected layers. a negative log-likelihood loss (NLLLoss) was used to train this network:

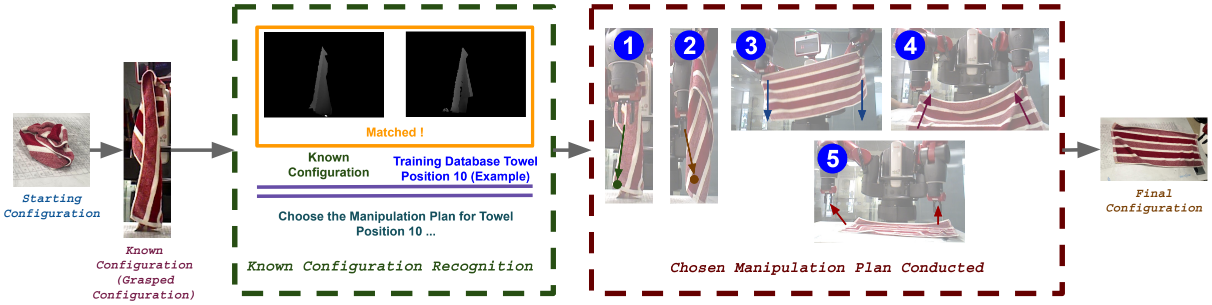

Starting Configuration
Known Configuration (Grasped Configuration)
Matched !
Known Configuration
Training Database Towel Position 10 (Example)
Choose the Manipulation Plan for Towel Position 10 ...
Known Configuration Recognition
1
2
3
4
5
Chosen Manipulation Plan Conducted
Final Configuration

Figure 6.2: *Known Configurations* Pipeline: a towel with a crumpled configuration (starting configuration) is grasped by a robot from a grasping point (the first grasping point as described in Section 6.3), giving rise to a *known configuration*. The *known configuration* is recognised, and a pre-designed five-step manipulation strategy (described in section 6.3) can be chosen based on the recognised *known configuration*. The robot manipulates the towel with the chosen manipulation strategy (described in section 6.3). A manipulation strategy is based on the idea that manipulating points can be located by recognising the known configuration of a garment. The manipulation strategy is approaching and manipulating the garment from the located manipulating points.

$$L(\theta) = \sum_{i=1}^{n} (y_i \log \hat{y}_{\theta,i} + (1 - y_i) \log(1 - \hat{y}_{\theta,i})) \tag{6.1}$$

where θ is the weight parameters of the network, y_i is the ground truth probability that the ith data point is positive, and $\hat{y}_{\theta,i}$ is the predicted probability that the ith data point is positive. Figure 6.2 shows how KCNet works in the *known configuration* pipeline. Figure 6.3 demonstrates the details of the proposed network.

As stated in section 6.3, a *known configuration* highly depends on the grasping points, therefore *known configurations* were labelled in terms to its corresponding grasping points. *Known configurations* should be recognisable by KCNet; thus they should be distinguishable from each other. Therefore, a *known configuration* represents all grasping points in a segmented area of a garment rather than a single grasping point. Discretising garments into different segments and defining a grasping point was proposed to represent one segment. Figure 6.4 shows ten grasping points on garments (jeans, shirts, sweaters, towels and tshirts). The experiments found that dividing a garment into ten segments can provide the most *known configurations* that are recognisable. Each segment has its corresponding grasping point as the centre of the area.

SOTA focuses on real-time policy updating in robotic deformable object manipulation. A robot will update its manipulation strategy after observing a new object state. The updates are computationally costly and time-consuming. In contrast, designing manipulation strategies in advance was proposed to avoid updating manipulation strategies to speed up task executions.

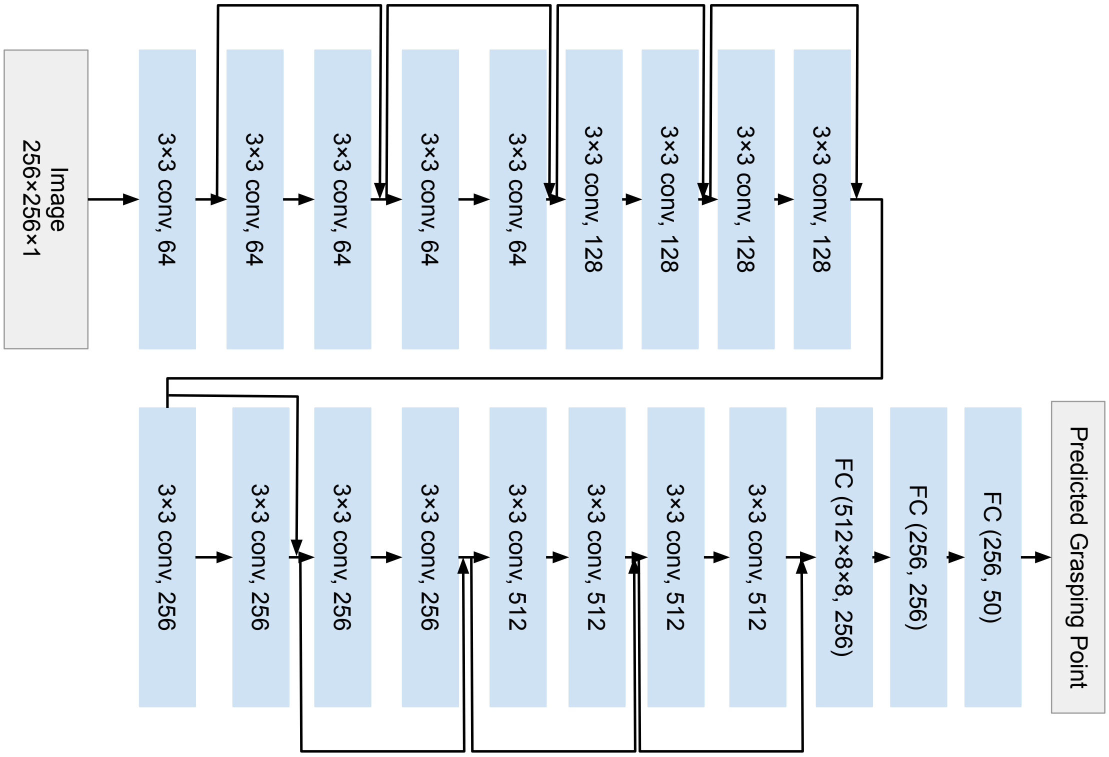


Figure 6.3: *The details of the proposed network*

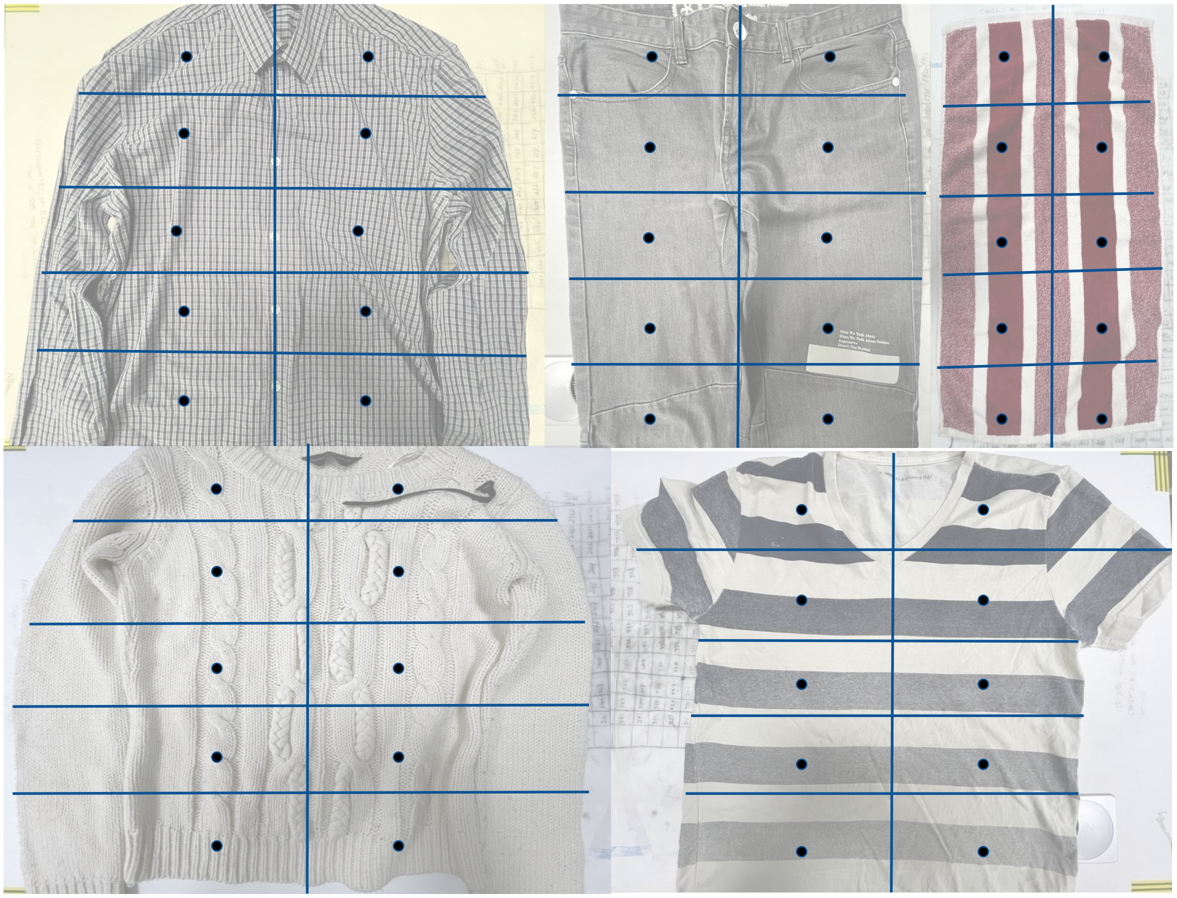


Figure 6.4: *Grasping Points Definition*: Each garment (jeans, shirts, sweaters, towels and tshirts) is divided into ten areas, and the centre of each area represents a grasping point (the first grasping point). The robot grasps the towel from these grasping points to form distinguishable *known configurations*.

A manipulation strategy comprises sequences of 6D poses and the robot's gripper state (i.e. open or close). As described in section 6.3, a proposed manipulation strategy includes finding the second and third grasping points, stretching to flatten garments and lifting garments down to the table. The proposed manipulation strategy design is thus based on these steps and ensures that each step requires the least actions. These sequences in CSV files were defined to transfer manipulation commands to the robot after a *known configuration* is recognised.

6.4 Experiments And Results

6.4.1 Experimental Methodology

The proposed *known configurations* pipeline consists of three stages: the robot recognises the *known configuration* of a grasped garment, based on the recognised *known configuration*, the robot chooses a pre-defined manipulation strategy, and then the robot flattens the garment.

To recognise garments' *known configurations*, a novel dataset was captured comprising depth and RGB images of garments from five categories: jeans, shirts, sweaters, towels and t-shirts. Each category has four garment instances, and as described in section 6.3, each garment instance has ten grasping segments represented by ten grasping positions. 100 images were collected for each position, where each image captures the *known configuration* of a garment grasped from a specific position. A total of 19,269 depth and RGB images were captured, respectively, with a resolution of 256×256 pixels. Figure 6.5 shows examples of images in this constructed dataset. An Xtion camera (introduced in Chapter 2 section 2.1.1) has been used to capture the RGB and depth images of garments. For each garment category, 10 manipulation strategies were pre-defined for each of the ten segments as described in section 6.3. Therefore, there are 50 manipulation strategies.

With this constructed dataset, a KCNet was trained which is a classification network that consists of a pre-trained ResNet 18 structure and a fully connected network. The learning rate is set to 10^{-3} and is decayed during the training with a step scheduler (a decay factor of 0.1 and a step size of 8).

A k-fold cross-validation approach (k-fold CVA) has been implemented to train and test KCNet, rather than the traditional approach of train-validate-test splits. The *k* value in the experiment is set to four, which means that the database is split into four groups for the k-fold CVA training and testing sessions. There are four garment instances in a group. Three groups are assigned as training groups for each session and one group as a testing group. All garment categories (jeans, shirts, sweaters, towels and tshirts) are trained together rather than trained separately in the experiment. The garment samples from the testing group are 'unseen' by KCNet. Classification accuracies were averaged, and the average classification accuracy was used as the classification accuracy for KCNet.

The robot in this experiment is a Baxter dual-arm robot with a table at the front to place the garments. A computer with Ubuntu 16.04 and an NVIDIA 1080 Ti GPU was used to train the KCNet. The robot is controlled by the Robot Operating System (ROS), and an Xtion camera captures images. In section 6.4.2, an example of a five-step manipulation strategy is demonstrate based on the recognised *known configurations* of a towel for flattening the towel. Improving pre-designed manipulation is planned in future research.

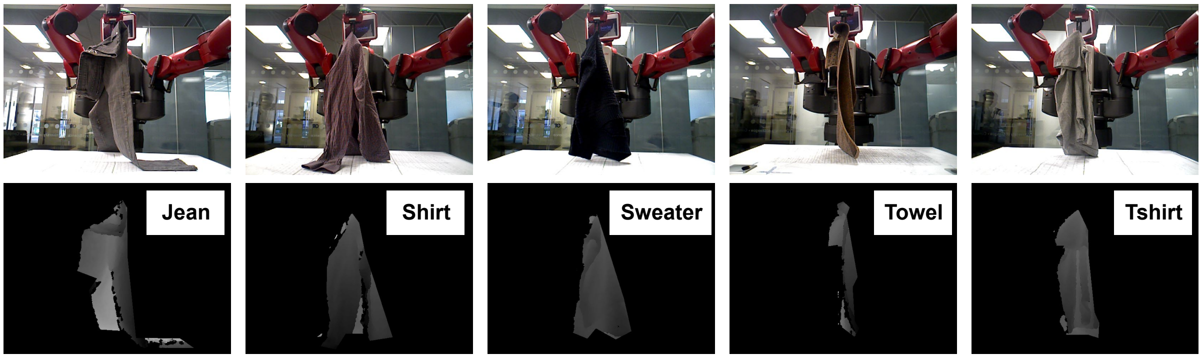


Figure 6.5: *Database Examples*: There are five garment categories in the database: jeans, shirts, sweaters, towels and t-shirts. The database contains masked depth and RGB images of four garment instances in each category, totalling five categories (Top: original images, Bottom: masked depth images).

6.4.2 KCNet Validation Results and Manipulation Demonstration

The performance of KCNets trained on RGB, depth and RGBD images was compared. Table 6.1 shows the results for each category in each k-fold cross-validation described in section 6.4.1. Table 6.1 shows classification accuracies that represent the percentages of *known configurations* recognised by the KCNet.

It is found that training a KCNet on depth images (89%) outperforms a KCNet on RGB images (73%). Depth images capture structural and physical characteristics of garments (as found in [134]), and therefore, enable a KCNet to recognise the *known configurations* of unseen garment instances. The RGB images capture texture characteristics that depend on the lighting condition

Table 6.1: k-fold cross validation experiment results: comparison between depth, RGB and RGBD images (unit: %)

Depth	1	2	3	4
towel	94.4	96.4	93.1	86.2
t-shirt	86.8	87.2	96.3	94.7
shirt	78.2	80.3	75.9	92.5
sweater	78.4	85.4	87.0	86.2
jean	99.3	95.8	95.3	99.1
average	*87.0*	*89.0*	*89.0*	*92.0*
AVERAGE: ***89.0***				
RGB	1	2	3	4
towel	67.0	55.9	74.1	64.5
shirt	87.9	58.4	75.4	91.4
t-shirt	70.8	71.2	72.1	76.9
sweater	54.6	42.0	68.1	85.2
jean	76.4	87.8	87.0	98.8
average	*73.0*	*62.0*	*75.0*	*83.0*
AVERAGE: **73.0**				
RGBD	1	2	3	4
towel	71.8	79.7	85.2	76.7
shirt	87.9	58.4	68.8	91.9
tshirt	74.6	68.3	84.4	83.9
sweater	52.7	51.6	74.7	77.0
jean	87.5	92.0	97.2	99.5
average	*76.0*	*70.0*	*82.0*	*86.0*
AVERAGE: **78.5**				

and shadows which hinders the ability of KCNet to predict *known configurations*. Meanwhile, it can be observed an increase from 73% to 78.5% for the KCNet trained on RGBD images (RGBD images are combined by concatenating the channels of RGB and depth images and used to train a KCNet.). However a significant gap of the KCNet trained only on depth images (89.0%). Depth information facilitates a KCNet to recognise the *known configurations* of unseen garments, while RGB information may affect the effectiveness of depth information.

Based on this ablation study, the KCNet trained on the depth images was chosen to conduct the manipulation experiments. Figure 6.6 shows an example of manipulations in the experiments. As described in section 6.3, the robot firstly recognises the *known configurations* of garments, then chooses manipulation strategies based on the recognised *known configurations*, and finally flattens the garments with the chosen manipulation strategies. It can be observed that the towel shown in Figure 6.6 is correctly flattened from its crumpled configuration.

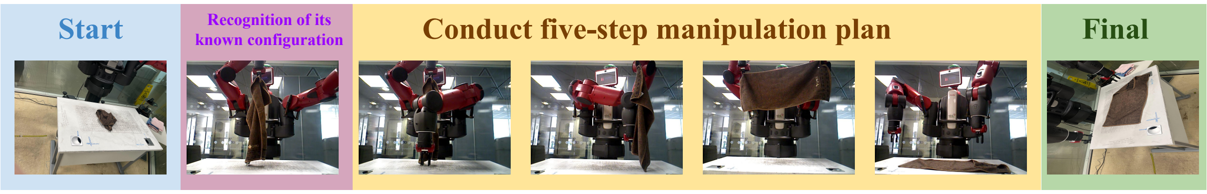

Start
Recognition of its known configuration
Conduct five-step manipulation plan
Final

Figure 6.6: *An Example of Manipulations*: an example is demonstrated: firstly, the robot grasps the towel from the table to recognise its *known configurations*. Then a manipulation strategy is chosen. Finally, the robot flattens the towel with the chosen manipulation strategy.

6.5 Discussion

This chapter has proposed a robotic garment flattening pipeline by recognising garments' *known configurations* and choosing pre-designed manipulation strategies to flatten garments. This chapter introduces the first part of this pipeline: how to recognise the 'known configurations' of garments. The *known configurations* can be recognised by taking advantage of gravity and reducing deformable objects' complexity. After the robot has grasped the garment, various and complicated initial configurations are converted into simple *known configurations* by taking advantage of gravity. There are several limitations in SOTA: human assistance is needed to ensure that the robot can stable manipulate garments. Human assistance means that the proposed network is not fully automatic. Thus Chapter 7 found an approach to solve the problems of human assistance by introducing hand-eye calibration with point clouds.

6.6 Conclusion

An effective robotic garment flattening pipeline has been proposed and consists of recognising the *known configurations* of garments and choosing pre-designed manipulation strategies to flatten them. The proposed pipeline also features training KCNets on depth images of real garments, resulting in higher recognition accuracy compared with previous work [6].

In the experiments, however, a robot can only recognise the *known configurations* of the garments with shapes from the five categories, while it is unable to recognise these garments with an unknown category. Future work aims to devise a continual learning framework so that a robot learns unknown shapes when it deals with these shapes. Additionally, It is believed that a robot can benefit from prior knowledge of the shapes [147] or physics properties [134] of garments to recognise *known configurations*, thus learning that prior knowledge during the robot's grasping garments and improving recognition accuracy on *known configurations* is proposed.

This chapter addressed the limitations of model-based and reinforcement learning approaches in Sections 2.3.1 and 2.3.2 and avoids grasping-and-re-grasping operations in Section 2.3.3 of Chapter 2. However, human assistance is needed in the proposed pipeline. Chapter 7 improved the performance of KCNet by using the prior knowledge of garment shapes.

This chapter has verified the following hypothesis:

A robot can pick up crumpled garments from any location, recognise their *known configurations*, and select pre-designed manipulation strategies based on recognised *known configurations*, where recognition accuracy is at least 80%.

This chapter proposes learning and recognising the *known configurations* of garments to select pre-designed manipulation strategies to flatten these garments, demonstrating effectiveness and efficiency from the results. However, the proposed pipeline needs human assistance, indicating that this pipeline is not fully automatic. This limitation is to be resolved in Chapter 7, which also improves garment *known configuration* recognition by incorporating the prior knowledge of garment shapes.

Chapter 7

A Data-Centric Approach For Dual-Armed Robotic Garment Flattening

Although Chapter 6 proposes a robotic garment flattening pipeline to recognise the known configurations of garments, there are two limitations. The first limitation is that the robot sometimes misrecognised shirts and sweaters. The second limitation is that human assistance is needed for flattening garments. In this chapter [1], *the garment similarity network (GarNet) introduced in Chapter 4 is used to improve the performance on the known configuration recognition. Also, this chapter introduces hand-eye calibration with point clouds to resolve the limitation of human assistance and shows the importance of learning garment shapes (geometric properties) of garments in robotic garment manipulation. This chapter finalises this thesis by providing a full and effective robotic garment flattening pipeline.*

7.1 Introduction

Due to the high dimensionality of object states, a garment flattening pipeline requires recognising the configurations of garments for a robot to select manipulation strategies to flatten garments. In this chapter, a data-centric approach was proposed to identify the *known configurations* of garments based on a *known configuration* network (KCNet) trained on depth images that capture the *known configurations* of garments and prior knowledge of garment shapes. The *known configurations* of garments are the configurations of garments when a robot hangs garments in the middle of the air. It is found that it is possible to achieve 92% accuracy when the

[1] *This chapter has appeared in [21]. Li Duan is the first author and the main contributor to this paper.*

robot recognises the common hanging configurations (the *known configurations*) of garments. This chapter also demonstrates an effective robotic garment flattening pipeline with the proposed approach on a dual-arm Baxter robot. The robot achieved an average operating time of 221.6 seconds and successfully manipulated garments of five different shapes.

7.2 Objectives and Motivation

Deformable objects have three characteristics: the dimensionality of their object states (configurations) is high, changes in their object states (configurations) are instant, and deformation patterns are irregular when a robot manipulates deformable objects. When deformable objects such as garments are flattened, folded, or gripped, their deformation patterns are irregular, making it challenging to find manipulating points. Researchers traditionally consider two ways to handle these challenges: model-centric and data-centric approaches. Model-centric approaches [66, 73, 82] focus on finding and defining specific models for objects, and manipulation strategies are derived and updated by monitoring changes in configurations of these models. Data-centric [9, 84, 97] approaches are divided into two categories. First, deformation patterns are learned from large-scale datasets, which are used for identifying grasping and manipulating points on garments. These points are then used for flattening or operating other manipulations on garments. Second, robots are trained to learn skills for manipulating objects via reinforcement and imitation learning.

There are several problems with model-centric approaches. Firstly, defining or finding black-box models that represent deformable objects is challenging. Deformable objects tend to have an infinite number of states, which means single or multiple black-box models cannot represent all possible object states. Model-centric approaches are usually validated in objects with simple deformations such as sponges [148], towels [8] or fruits[66]. Complex, deformable objects such as garments are rarely employed because black-box models for representing object states would require an almost infinite database representing all possible object states of these deformable objects. Secondly, models cannot be generalised, which means changing objects means updating models. Different objects have different characteristics (e.g. materials and textures), which means that black-box models are specific and object-restricted. A new model is needed for a new object, while knowledge or parameters may be invalidated, causing robotic manipulation tasks to be constrained to specific objects.

Similarly, limitations exist for data-centric approaches. Robots are usually trained in simulated environments for reinforcement and imitation learning approaches due to limitations in real

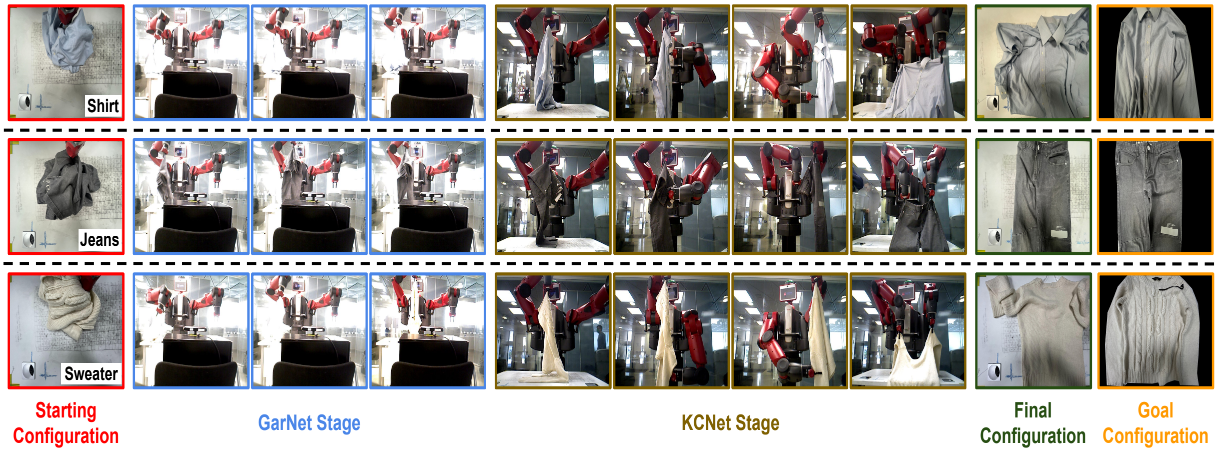


Figure 7.1: *Robotic Demonstration Examples*: Garments with crumpled starting configurations are firstly recognised in their shapes by GarNet, and the Baxter robot recognises their *known configurations* with KCNet from the depth images of their *known configurations* and prior knowledge of their shapes. The robot uses pre-designed manipulation strategies to flatten these garments (the pre-designed grasping points are fine-tuned by the cloud-point method described in section 7.3.5). From *top* to *bottom*: a shirt, a jean, a sweater

environments. Robots may be rewarded with actions that are not feasible in real environments due to the range of constraints of orientations and positions for robotic grippers [15]. Robots can take 100-200 epochs to be trained, which would take a significant time if trained in real environments [92]. While reward policies trained in simulated environments can potentially be invalid in real environments due to the differences in external factors such as illumination conditions and the imperfections of simulated objects [99].

In this chapter, a data-centric approach has been proposed that aims to find *known configurations* of garments and a robot that flattens garments with pre-designed manipulation strategies. *Known configurations* of garments are the configurations of hanging garments after a robot has grasped them from a given, random grasping point. A *known configuration* network (KCNet) has been trained in this paper to recognise the *known configurations* of garments with five different shapes. Compared with model-centric approaches, the proposed approach does not require defining specific models for garments, and the robot is not trained to learn manipulation skills with reinforcement or imitation learning. Instead, a robot selects pre-designed manipulation strategies based on the recognised *known configurations* of garments. Traditional approaches such as [8] update manipulation strategies by monitoring changes in the configurations of deformable objects in real-time using finite element methods, which is computationally expensive. Using selected pre-designed manipulation strategies, which will not be updated during manipulations, based on their recognised *known configurations* is effective and fast.

Chapter 6 successfully demonstrated the effectiveness of learning *known configurations* for garment flattening tasks. However, it is found that the KCNet often recognised shirts' *known configurations* as sweaters ones. It is believed that the problem happened because the KCNet did not have any prior knowledge about the shapes of garments. Thus it associated shirts with sweaters. Therefore, in this chapter, a robotic garment flattening pipeline is proposed, where a robot recognises the shapes of garments first and identifies the *known configurations* of garments based on the recognised shapes. This chapter also resolved the limitation of human assistance in Chapter 6 by introducing hand-eye calibration with point clouds. The hypothesis in this chapter is that *A robot can flatten crumpled garments by predicting the shapes of garments, recognising the known configurations of garments based on predicted garment shapes, and selecting manipulation strategies to flatten garments, where recognition accuracy is at least 90%, and flattening garments require less than 250 seconds.* The contributions are as follows:

- A robotic garment flattening pipeline has been proposed, where a robot predicts garment shapes by continuously perceiving garments being grasped. Then, it recognises the *known configurations* of garments based on the predicted shapes and selects pre-designed garment manipulation strategies to flatten these garments. Examples of successful manipulations can be found in Figure 7.1;
- The proposed pipeline features an on-the-fly strategy. Traditional approaches [113, 23, 106] focus on accurately grasping points by learning features of different parts of garments in real-time. While in this pipeline, recognising the *known configurations* of garments makes it possible only to fine-tune the pre-designed grasping points to locate grasping points on the garments.
- The proposed pipeline does not require sophisticated modelling for garments, as the *known configurations* of garments were learned by training a *known configuration* network from a *known configuration* dataset (i.e. a model-free and data-driven approach), resolving the challenges of modelling garments.

The methodology is introduced in section 7.3, experiment setup in section 7.4 and experiment results in section 7.5.

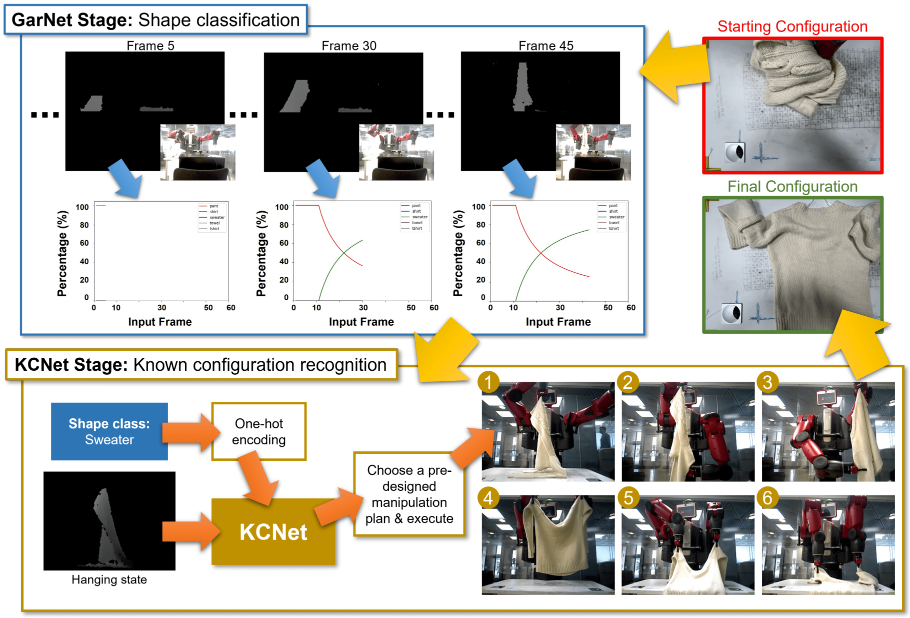


Figure 7.2: *Full pipeline*: A sweater with a crumpled starting configuration is randomly grasped by the robot to a point above the table. The robot continuously perceives the sweater's motion trajectory to gain confidence in predicting the sweater's shape. After, the robot recognises the *known configuration* and then uses a pre-designed three-step manipulation strategy to flatten the sweater. The pre-designed first and second grasping points are fine-tuned based on the closest point from the pre-defined grasping point and the hanging sweater.

7.3 Materials and Methodology

7.3.1 Flattening pipeline

The flattening task in this paper is defined as the robot sliding the garment over the edge of a table after the garment has been unfolded in the air based on prior knowledge of its shape and *known configuration*. Therefore, the proposed flattening pipeline consists of three core stages: shape prediction, known configuration prediction and manipulation stages as shown in Fig. 7.2.

The flattening starts with a robot grasping a garment from a table to a point above the table (approx. 1m from the table). During the motion trajectory, the robot predicts the garment's shape by continuously perceiving it as it is being translated to estimate its shape. For this, a garment similarity network was employed (ref. 7.3.2). After the robot successfully predicts the garment shape, it predicts the garment's *known configuration* (ref. Section 7.3.3) using the depth image of the garment hanging state and the prior knowledge of the garment shape as inputs. A pre-designed manipulation strategy matched with the recognised *known configurations* is used to flatten garments (ref. Section 7.3.5).

7.3.2 Garment's shape prediction

The first part of the proposed robotic garment flattening pipeline is about predicting the shapes of garments. The prior knowledge of the shapes of garments facilitates recognising the *known configurations* of garments for matching manipulation strategies to flatten garments. In the experiments, the garment similarity network (GarNet) in Chapter 4 has been employed to predict garment shapes. Compared with traditional approaches of using single images to predict garment shapes, GarNet enables a robot to continuously perceive the video frames of garments being grasped from a table to a point above the table.

Specifically, the robot grasps a garment from a table to a point above the table (approx. $1m$ above the table). A sequence of depth images (as captured while the robot moves the garment from the table) are input into GarNet, which maps each image into a Garment Similarity Map (GSM) defined as a Garment Similarity Point (GSP). Each garment shape learned by GarNet is clustered on the GSM, and the GSP from the input depth image is mapped into one of these shape clusters. After the robot perceives at least 20 images, if 80% of GPSs from the input images belong to the same shape, the shape cluster class is used as the predicted shape for the grasped garment. Further details and experiments can be found in Chapter 4. A GarNet trained on depth images has been used based on the experiment results from Chapter 4.

7.3.3 Robotic garment flattening with known configuration network (KCNet)

After garment shapes are successfully predicted by GarNet in section 7.3.2, the robot will recognise the *known configurations* of garments and match pre-designed manipulation strategies with the recognised *known configurations* to flatten garments. Figure 7.2 shows the pipeline of the proposed robotic garment flattening pipeline.

A *known configuration* is defined as a garment configuration in a hanging state after being grasped from a different configuration. That is, the *known configurations* of garments highly depend on grasping points because of gravity, regardless of their starting configurations, which means that the irregular and complex starting configurations of garments are converted into stable, constant *known configurations*.

For this, KCNet (introduced in Chapter 6) has been adopted such that it can recognise the *known configurations* of garments based not only on images of their *known configurations* but also on prior knowledge of garment shapes, which GarNet predicted (Section 7.3.2). In conclusion, KCNet in this chapter has been improved by introducing the prior knowledge on garment shapes, which facilitates the network to recognise the *known configurations* of garments.

KCNet consists of a ResNet18 convolutional network and three linear networks. A KCNet consists of a ResNet18 convolutional network [122] and three linear networks. An image of an *known configuration* is an input to the KCNet to get its extracted features. The garment shape, as predicted by GarNet is encoded to a one-hot vector and then input into a linear network to get its latent representation. The feature extracted from the image and the latent representation from the one-hot vector of the predicted shape is concatenated, which is then input into a linear network to output the *known configuration* of the garment. Similar to Chapter 6, the KCNet in this chapter is trained with a Negative Log-Likelihood Loss (NLLLoss). Figure 7.3 demonstrates the details of the proposed network.

7.3.4 Pre-designed manipulation strategies

After the *known configurations* of garments are successfully recognised from KCNet, the robot matches pre-designed manipulation strategies with the recognised *known configurations*. A pre-designed manipulation strategy is a sequence of end-effector commands for the robot to flatten a garment. Each gripper command contains 16 parameters, including XYZ coordinates of the right and left grippers, orientations (defined as a quaternion) of right and left grippers, a choice of grippers (left or right) and gripper status (open or close). On average, 18 gripper robotic actions are needed to flatten a garment.

Similar to [23] [24], a manipulation strategy follows a three-step manipulation rule: find the first grasping point, the second grasping point, and stretch and flatten garments. A *known configuration* has a corresponding pre-designed manipulation strategy. Ten pre-designed manipulation strategies have been designed for each shape of garments, totalling 50 manipulation strategies for the five shapes in the experiments.

7.3.5 Fine-tuning Garment Grasping

Grasping points are pre-designed in manipulation strategies, consisting of the first and second grasping points. However, garments deform irregularly; thus these pre-designed grasping points must be fine-tuned for stable grasps during a flattening operation. Therefore, locating grasping

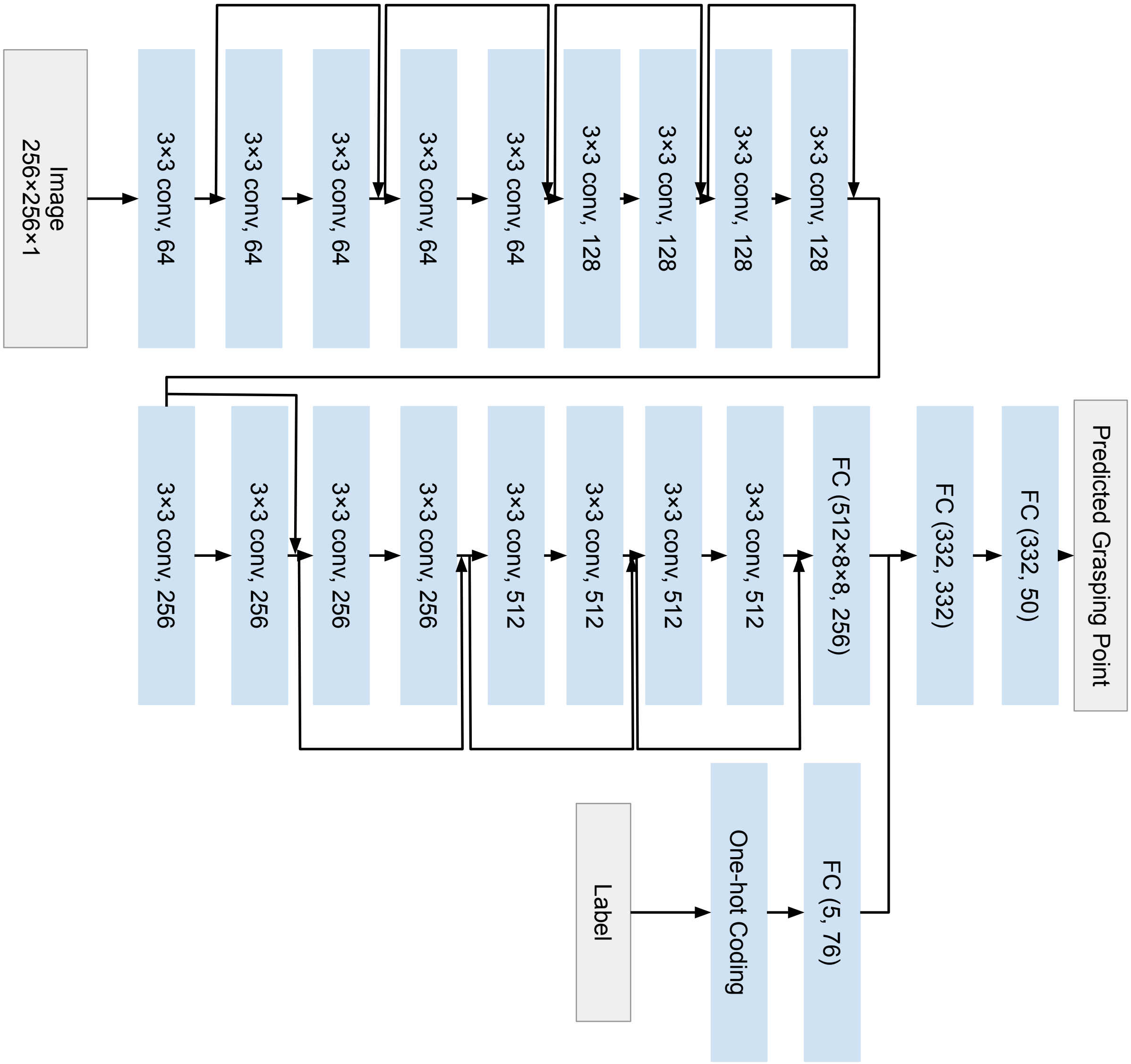


Figure 7.3: The details of the improved KCNet

points by finding the closest point between the robot's gripper and the garment in point clouds has been proposed.

Firstly, the Xtion camera is placed in front of the robot for recognising the *known configurations* of garments. Khan's *et al.* [149] hand-eye calibration approach has been used to bring the camera and the robot into alignment. After the robot and camera coordinate systems are aligned, point clouds of garments are captured. From the captured point clouds, the points belonging to garments have a specific range of values because they have similar distances to the camera. Therefore, points belonging to garments can be easily identified and segmented from the points belonging to other objects (for example, the Baxter robot). Then, points that have the smallest distances with pre-designed grasping points were found and defined as fine-tuned grasping points used for grasping garments. Therefore, the grasping points can be ensured to be on garments and stable grasps can be generated on-the-fly in this way.

In the experiments, there is no need to identify local features or landmarks of garments as used in previous research ([113, 23, 106]); visual servoing is only needed when *known configurations* are recognised. Therefore, the proposed data-centric approach does not require constructing models for garments and updating manipulation strategies for unseen garments. Figure 7.4 shows an example of pre-designed grasping points (coloured in yellow) and fine-tuned grasping points (coloured in green) for the first and second grasping points (described in section 7.3.4) of a sweater. It can be observed that the pre-designed grasping points are near to the garment but the fine-tuned grasping points can ensure stable grasps compared with pre-designed grasping points.

7.3.6 Robotic Setup

A Baxter dual-arm robot is employed for in the experiments, controlled by MoveIt software. A four-legged table is placed in front of the Baxter robot to place garments in front of the robot. Two Xtion stereo cameras are set up facing the Baxter robot at two locations to enable wide- and narrow-field image capturing. Both cameras capture RGBD images (i) for predicting garment shapes and (ii) for recognising garment *known configurations* and fine-tuning pre-designed points (Section 7.3.5). For predicting garment shapes, the garment similarity network (Section 7.3.2) needs to observe how garments deform during grasping. Therefore, the camera must capture garment deformations as the robot grasps it up; hence, this camera is placed at a larger distance from the Baxter robot. Conversely, for recognising the *known configurations* of garments (Section 7.3.3), the camera needs to capture the details of garments and is, thus, placed close to the Baxter robot. the manipulation strategies were implemented using the robot operating

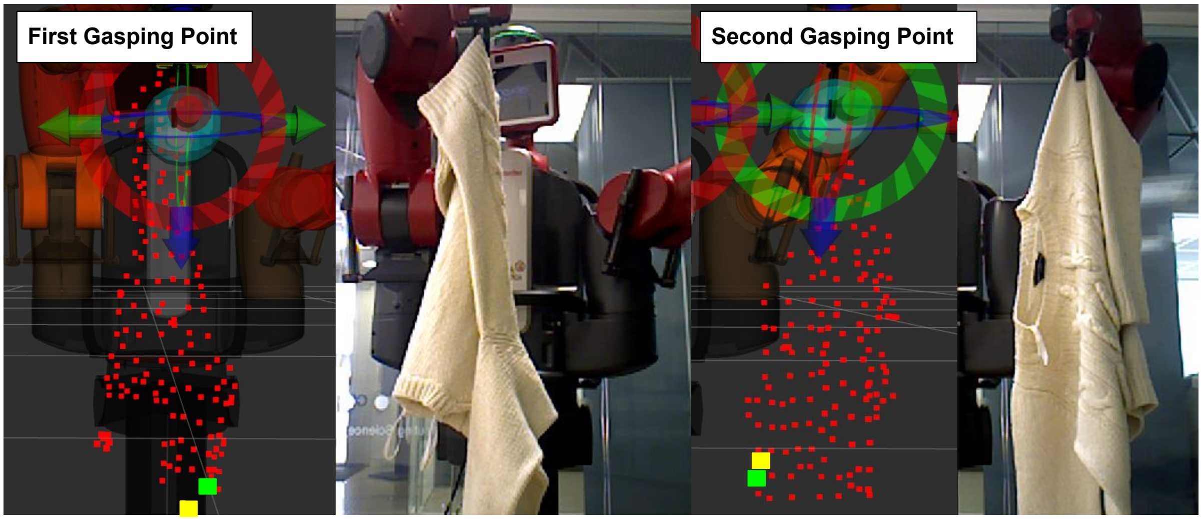


Figure 7.4: *Grasping Points Location with Point Cloud*: the pre-designed grasping points need to be fine-tuned to ensure stable robotic grasps. A point cloud is used to fine-tune the pre-designed grasping points. (*Top*: fine-tuning the pre-designed first grasping point of a sweater; *Bottom*: fine-tuning the pre-designed second grasping point of the sweater; *Yellow*: the pre-designed grasping points; *Green*: the fine-tuned grasping points)

system (ROS, Kinect version) and the Ubuntu 16.04 environment, installed on a computer with a Core i7 CPU and an Nvidia 1080 Ti GPU. An *openni2* [2] camera-controlling system controls the cameras. The sampling rate is 30 Hz for both cameras.

Five shapes of garments were used with four different instances for each shape, totalling 20 garments tested. The shapes considered in this chapter are jeans, shirts, sweaters, towels and t-shirts. Garments have different colours, textures and materials. Jeans are made of denim, shirts are made of polymer, and sweaters, t-shirts and towels are made of cotton. Experiments are conducted in a robot laboratory, where lighting conditions are set the same during each experiment. The laboratory is enclosed, which means that the external environment is stable and does not affect the experiments.

[2] `http://wiki.ros.org/openni2_launch`

7.4 Experiment Setup

7.4.1 Database

The database constructed in Chapter 6 has been used. For each grasping point (equivalent to each *known configuration*), 100 RGB and depth images were captured. For each garment instance, the robot randomly grasps the garment and lifts it above the table ten times, totalling 19,269 images in the database. Note that 731 images had image artefacts or noise. Thus, they were removed from the database. Each image has a resolution of 256×256.

7.4.2 KCNet training experiments

As described in 7.3, KCNet consists of ResNet18 and three linear networks, where the features of predicted shapes and the images of *known configurations* are concatenated and used for recognising *known configurations*.

A leave-one-out cross-validation method was implemented in the experiments to validate the proposed approach. The proposed KCNet is trained on a small database. Thus, implementing the leave-one-out cross-validation method ensures that the KCNet has a robust performance in recognising the *known configurations* of garments. The database is divided into training and testing groups. The images from three of four garment instances for each shape serve as a training group, while the images of the remaining instance serve as the testing group. It is ensured that the garments in testing groups never appear in training groups, which means testing garments are unseen by GarNet and KCNet. Therefore, four experiments have been conducted, where testing and training groups differ each time. The recognition accuracies are averaged across the four experiments, and the averaged value is used as the accuracy of the KCNet. KCNet combined with GarNet in this chapter has been compared with the KCNet in Chapter 7. An ablation study was conducted on recognition accuracy on depth, RGB, and RGBD images to investigate image formats' impacts on recognising *known configurations*. The KCNet combined with GarNet with the highest accuracy has been used for the robotic garment flattening pipeline. an Adam optimiser was used to train KCNet, which is regulated by a learning-rate scheduler with a step size of eight epochs and a decay factor of 10^{-1}. The initial learning rate for the optimiser is 10^{-3}, the number of training epochs is 30, and the batch size is 64.

7.4.3 Robotic Garment Flattening Pipeline

20 garment flattening operations were performed to validate the robotic garment flattening pipeline experimentally. For this, starting configuration states, final configuration states, times for each manipulation step, and success rates (the percentage of successful tests in all tests) were recorded. A starting/final configuration state is computed as:

$$Starting\ State = \frac{S_{start}}{S_{goal}} \times 100\% \tag{7.1}$$

$$Final\ State = \frac{S_{ending}}{S_{goal}} \times 100\% \tag{7.2}$$

S_{start} refers to the area in pixels of the starting configurations of garments, and S_{ending}, the area in pixels of the final configurations of garments. S_{goal} represents the area in pixels of the goal configurations of garments. A goal configuration is a garment configuration when the garment is fully flattened (referenced in [93]). An example of goal configurations is shown in Figure 7.1. To capture the starting, final and goal configurations, a camera was positioned at a fixed distance above the table, captured images at each state and cropped them using the table's corners and edges as guidelines.

7.5 Experiment Results

7.5.1 KCNet training and ablation study results

This experiment has two objectives: investigating the influence of image formats on the recognising accuracy of KCNet and comparing KCNet with the prior knowledge of garment shapes with KCNet without this information.

Tables 7.1 and 7.2 show the results for unseen garment instances of KCNets trained on different image types. It can be observed that the KCNet trained on depth images (92% for the KCNet with prior knowledge of garment's shapes and 89% for the KCNet without prior knowledge of garment's shapes) has the best performance compared to RGB (78% with prior knowledge of garment's shapes and 73% without) and RGBD (81.5% with prior knowledge of garment's shapes and 78.5% without).

Table 7.1: KCNet ablation study: KCNet without prior knowledge of the garment's shape (Accuracy, unit: %)

SHAPE	Exp 1	Exp 2	Exp 3	Exp 4
towel	94.4	96.4	93.1	86.2
t-shirt	86.8	87.2	96.3	94.7
shirt	78.2	80.3	75.9	92.5
sweater	78.4	85.4	87.0	86.2
jean	99.3	95.8	95.3	99.1
average	*87.0*	*89.0*	*89.0*	*92.0*
	Depth AVERAGE: *89.0*			
SHAPE	Exp 1	Exp 2	Exp 3	Exp 4
towel	67.0	55.9	74.1	64.5
t-shirt	70.8	71.2	72.1	76.9
shirt	87.9	58.4	75.4	91.4
sweater	54.6	42.0	68.1	85.2
jean	76.4	87.8	87.0	98.8
average	*73.0*	*62.0*	*75.0*	*83.0*
	RGB AVERAGE: *73.0*			
SHAPE	Exp 1	Exp 2	Exp 3	Exp 4
towel	71.8	79.7	85.2	76.7
t-shirt	74.6	68.3	84.4	83.9
shirt	87.9	58.4	68.8	91.9
sweater	52.7	51.6	74.7	77.0
jean	87.5	92.0	97.2	99.5
average	*76.0*	*70.0*	*82.0*	*86.0*
	RGBD AVERAGE: *78.5*			

Depth images capture structural and spatial information about garments, compared to RGB images which capture texture information. Structural and spatial characteristics are similar between garments of similar shapes, while textures are easily affected by lighting conditions and the garment's colours that vary across different garments.

Also, compared with KCNets without prior knowledge of garment's shapes [20], KCNets with prior knowledge of garment's shapes demonstrate better performance on all types of images, specifically on recognising *known configurations* of shirts and sweaters. That is, shirts and sweaters share similar structures; thus, identifying differences between their *known configurations* based only on their depth images is challenging. The latter is overcome by having prior knowledge of garments' shapes. From the results of this experiment, KCNet trained on depth images with the prior knowledge of garment shapes has been selected for the following robotic garment flattening pipeline tests.

Table 7.2: KCNet ablation study: KCNet with prior knowledge of the garment's shape (Accuracy, unit: %)

SHAPE	Exp 1	Exp 2	Exp 3	Exp 4
towel	93.8	96.5	92.4	87.5
t-shirt	93.9	93.1	94.4	95.0
shirt	90.1	82.0	90.4	96.2
sweater	76.0	90.3	92.5	83.7
jean	98.5	94.2	94.5	98.9
average	91.0	91.0	93.0	92.0
	Depth AVERAGE: 92.0			
SHAPE	Exp 1	Exp 2	Exp 3	Exp 4
towel	67.2	78.5	85.8	63.9
t-shirt	84.6	71.7	73.5	79.8
shirt	90.5	68.1	72.8	95.2
sweater	54.0	56.3	81.2	71.0
jean	82.0	89.9	86.5	98.0
average	77.0	72.0	80	82.0
	RGB AVERAGE: 78.0			
SHAPE	Exp 1	Exp 2	Exp 3	Exp 4
towel	74.2	79.3	84.0	65.6
t-shirt	85.3	77.8	90.8	86.7
shirt	90.5	73.2	67.5	94.4
sweater	61.1	67.2	77.0	84.3
jean	90.4	87.5	95.0	98.5
average	81.0	77.0	82.0	86.0
	RGBD AVERAGE: 81.5			

7.5.2 Robotic Garment Flattening Results

Table 7.3 shows the starting configuration states, running times for each step, final configuration states and success rates. On average, the Baxter robot requires 29 seconds to predict the garment shapes with GarNet. For garment flattening, Baxter needs, on average, 79.8 seconds to grasp the first grasping point, 60.2 seconds to grasp the second grasping point, and 52.6 seconds to stretch and flatten a garment. Compared to the SOTA [8], where the robot needs approximately 15 minutes to fold a towel, the proposed approach has a faster manipulation operation. Traditional approaches update robotic manipulation strategies in real time, which requires costly computations. Also, updating robotic manipulation strategies may result in robots' unnecessary actions, taking additional time.

On average, the starting configuration states are 27.80% (ref. Section 7.4.3), while it increases to 80.1% for final configuration states. The results show that the robot successfully flattened garments and reached a high average final configuration state (80.1%) with the proposed approach. It is found that jeans, shirts, sweaters and t-shirts have better final configuration states than towels. When Baxter finds the second grasping point for flattening the towels, it sometimes grasps the incorrect corners because the towels are symmetrical, resulting in final configuration states lower than other garments. This chapter does not set pass and failure binary categories for the flatness test but quantitatively evaluates the coverage rates for flattened garments. Failure cases happen when the robot can not recognise garment shapes and *known configurations* with the GarNet-KCNet mechanism and when garments slip and drop from the robot. In future work, a small coverage rate for flattened garments can be improved by re-grasping and re-flattening the garments, while the small coverage rate will not be counted as a failure case.

Table 7.4 shows a comparison between the proposed approach with SOTA. Sun *et al.* [125] was one of the first to demonstrate dual-arm flattening of towels and achieved a dual-arm success rate of 46.3%; their approach required approx. 20 minutes to flatten a squared towel with one or two folds. Li *et al.* [23] and Martin *et al.* [150] achieved high manipulation success rates; however, they required re-grasping garments to ensure that the robot recognised the pose (equivalent to the *known configuration* in this chapter). Their approaches, therefore, required longer manipulation time. Li *et al.* [23] achieved a lower pose estimation accuracy than the proposed approach because they trained their network with simulated data. Martin *et al.* [150], Yan *et al.* [8] and Sun *et al.* [125] only tested towels in their experiments, while five different shapes of garments has been tested with the proposed approach. Overall, the proposed approach achieved the fastest manipulation and second highest *known configuration* recognition accuracy but has a lower success rate than other approaches.

The average success rate for the proposed approach is 66.7%. It can be observed that jeans and towels have the lowest success rates (57.1%). Jeans are heavier than other garments, thus they sometimes slip from the robotic grippers and result in failures. Towels are sometimes grasped from wrong grasping points because they are symmetric, resulting in failure cases. In future research, grippers suited for manipulating garments (e.g. [125]) are needed for grasping and flattening jeans and better manipulation strategies for grasping and flattening towels. Figure 7.1 shows three successful garment flattening operations of three shapes (shirt, jeans and sweater).

7.6 Discussion

In this chapter, an effective robotic garment flattening was demonstrated by firstly recognising garment shapes, then recognising the *known configurations* of garments with the recognised garment shapes and the depth images of garments' *known configurations* and finally using pre-

Table 7.3: Robotic Garment Flattening Results

SHAPE	**Starting State**	*Shape Prediction*	*Grasping Point 1*	*Grasping Point 2*
Jeans	**40.51%**	*26.0s*	*55.5s*	*50.0s*
Shirt	**27.04%**	*30.0s*	*71.0s*	*47.5s*
Sweater	**25.98%**	*28.8s*	*86.0s*	*62.0s*
Towel	**23.85%**	*30.0s*	*82.5s*	*71.0s*
T-shirt	**21.61%**	*30.0s*	*104.0s*	*70.5s*
Average	**27.80%**	*29.0s*	*79.8s*	*60.2s*
SHAPE	*Flattening*	*Total Time*	**Final State**	Success Rate
Jeans	*72.5s*	*204.0s*	**87.75%**	57.1%
Shirt	*42.0s*	*190.5s*	**96.30%**	66.7%
Sweater	*66.0s*	*242.8s*	**76.65%**	80.0%
Towel	*41.0s*	*224.5s*	**65.44%**	57.1%
T-shirt	*41.5s*	*246.0s*	**74.06%**	80.0%
Average	*52.6s*	*221.6s*	**80.10%**	66.7%

Table 7.4: Comparison with SOTA (N/A means that the value is not reported in the paper.)

Method	Estimation Accuracy	Running Time	Success Rate
Sun [125]	N/A	1,200	46.3%
Li [23]	83%	N/A	80.0%
Yan [8]	N/A	900s	N/A
Maitin [150]	**96%**	1478 s	**81%**
The proposed approach	92%	**221.6s**	66.7%

designed manipulation strategies to flatten garments. However, only garments from the five shapes can be manipulated in the experiments because manipulation strategies are designed based on the five shapes. Garments with an unknown shape (for example, shorts) can not be manipulated, limiting the proposed approach to specific garment shapes.

7.7 Conclusion

In this chapter, an effective robotic garment flattening pipeline has been proposed where a Baxter robot predicts garment shapes, recognises their hanging state (*known configurations* of garments) based on their shapes, and uses pre-designed manipulation strategies to flatten them. It is found that the proposed approach achieved an accuracy of 92% for recognising *known configurations*, and the robot takes, on average, 29 seconds to predict the garment shapes and 192.6 seconds on average to manipulate and flatten garments.

In this chapter, only garments from the five shapes were used because manipulation strategies are designed based on these five shapes. Garments with an unknown shape (for example, shorts) can not be manipulated, limiting the proposed approach to specific shapes of garments, and this is an open problem in deformable object manipulation. Implementing a reinforcement learning approach [132] can be applied to improve the success rate of garment flattening, where the robot learns basic skills which can be composed to manipulate any garment shape. Another limitation of the proposed flattening pipeline is that two cameras were used to recognise garments' shapes and *known configurations*. In future work, only one camera is needed for both shape and *known configuration* predictions and active vision approaches will be investigated to control the camera's viewpoint.

This chapter finalises this thesis by introducing an effective robotic garment flattening pipeline, which addressed the limitations described in Sections 2.3.1 2.3.2 and 2.3.3 in Chapter 2. This chapter resolved the human-assistance problem in Chapter 6 and improved KCNet performance by recognising the *known configurations* of garments with prior knowledge of garment shapes compared with Chapter 6. This chapter verifies the following hypothesis: A robot can flatten crumpled garments by picking them up from any location, predicting garment shapes, recognising the *known configurations* of garments based on predicted garment shapes, and selecting pre-designed manipulation strategies to flatten garments, where recognition accuracy is at least 90%, and flattening garments require less than 250 seconds with a success rate over 60%

This chapter introduced an effective robotic garment flattening pipeline, demonstrating the importance of incorporating garment shapes (geometric properties) in robotic garment manipulation. Also, this chapter resolves the limitation of human assistance in Chapter 6 by implementing hand-eye calibration with cloud points. However, this chapter does not incorporate garment physics properties in the proposed pipeline introduced in Chapter 5. It is believed that incorporating garment physics properties can further improve the performance of the proposed pipeline. The future work on garment physics property incorporation in the proposed robotic garment flattening pipeline (and other robotic garment manipulations) is discussed in Chapter 8.

Chapter 8

Conclusion and Future Work

Although this thesis addressed the limitations within SOTA discussed in Chapter 2, there is still future work that can improve and extend the research in this thesis. This chapter revisits the hypotheses and objectives, summarises the contributions and limitations of this thesis, and provides future work that can improve and extend the research in this thesis. This chapter revisits the objectives and hypotheses proposed in Chapter 1 in Section 8.1 and discusses how these hypotheses are validated from Chapters 3 to 7 in Section 8.2.4. This chapter also summarises the contributions of this thesis in Section 8.2. The limitations of this thesis are discussed in Section 8.3, and future work to address these limitations are also proposed in Section 8.3.

8.1 Objectives and Hypotheses Revisited

This thesis primarily targets three objectives:

- To predict shapes and visually perceived weights of garments by robots continuously perceiving video frames of garments being grasped to gain confidence;
- To predict physics properties of real garments and fabrics by learning physics similarities between simulated fabrics; and
- To devise an effective robotic garment flattening pipeline by priorly predicting geometric properties (shapes) of garments, recognising *known configurations* of garments based on predicted shapes of garments and selecting pre-designed manipulation strategies to flatten garments.

8.2 Summary of Contributions

Contributions from this thesis are categorised into three aspects: continuous-perception mechanism, prediction of physics properties of real garments and an effective garment flattening pipeline.

8.2.1 Continuous-perception paradigms for unseen garment shape and visually perceived weight prediction

This thesis proposed a continuous-perception mechanism, where the predictions of shapes and visually perceived weights of garments could be achieved by a robot continuously perceiving video frames of garments being grasped. This thesis proposed two approaches to realise this mechanism: a CNN-LSTM network with a “moving-average” strategy and a garment similarity network with an “early-stop” strategy. In Chapter 3, the robot perceived the entire video frames of garments being grasped and made decisions based on the values of their moving averages. In Chapter 4, the concept of garment similarity maps based on contrastive learning was proposed instead of using a traditional classification approach to investigate the continuous-perception mechanism. A garment similarity map encodes garment images into different clustering groups according to their shapes or visually perceived weights. When video frames of unseen garments are mapped on a garment similarity map, each video frame is given a label according to its decision-point position in the map. Shape or visually perceived weight categories with most labels within a clustering group on the garment similarity map are determined as predicted shapes or visually perceived weights. The experimental results showed that the garment similarity network outperformed the SOTA and requires less time because of the “early-stop” strategy. An “early-stop” strategy means that a robot does not need to perceive entire video frames to make decisions faster than the continuous perception approach [3] in SOTA.

8.2.2 Prediction of physics properties of real garments and fabrics via learning the physics similarities between simulated fabrics

This thesis describes an approach to predicting the physics properties of real garments and fabrics by learning physics similarities between simulated fabrics. Garments are usually complex deformable objects consisting of parts such as sleeves, collars, cuffs and pockets and from various colours and textures. These attributes make it difficult to simulate garments in a simulation engine and learn the physics properties of garments from their simulations. Also, garments in simulated environments is challenging because these parts of garments require a large number of meshes or other simulating tools to simulate, which is time-consuming. Fabrics can be easily modelled in simulated environments, which are used to match real garments or fabrics by minimising their "physical similarity distances" with a Bayesian optimiser. This thesis inspires the following researchers to use simple deformable objects to study the physics properties of complicatedly-structured deformable objects compared with SOTA.

8.2.3 An effective robotic garment flattening pipeline

This thesis introduced an effective robotic garment flattening pipeline by priorly predicting geometric properties (shapes) of garments, recognising *known configurations* of garments based on their shapes and selecting pre-designed manipulation strategies to flatten garments. Previous research on robotic garment manipulation often involves real-time updating of manipulation trajectory after observing a new object state of garments, which is computationally expensive and time-consuming. The proposed pipeline focused on offline manipulation strategy design, which does not require online updating and significantly boosts manipulation procedures. The proposed pipeline also prompted predicting geometric properties (shapes) of garments before recognising *known configurations* of garments because these properties are a primary factor that determines garment *known configurations*. This thesis found that prior knowledge of garment shapes (geometric properties) improves a robot's ability to recognise the *known configurations* of garments compared with SOTA.

8.2.4 Hypotheses – Revisited

This thesis validates the hypotheses in this section.

The first hypothesis is:

- A robot can predict the shapes and visually perceived weights of unseen garments by implementing a CNN-LSTM network to learn the deformations of grasped garments and making predictions based on moving averages across video frames of grasping garments, where accuracy is at least 10% better than using single images for predictions.

In Chapter 3, a robot makes decisions on shapes and visually perceived weights of garments by perceiving video frames. The probabilities of each frame belonging to each category are accumulated. Therefore decisions are made from whole video frames rather than single images. From the performance results demonstrated in Chapter 3, it can be observed that applying moving averages has increased the CNN-LSTM performance on visually perceived weights (discretised weights) of garments by 21.7% (from 48.3% to 60%), which shows the importance of the continuous-perception mechanism in garment visually perceived weight predictions.

The second hypothesis is:

- A robot can predict the shapes and visually perceived weights of unseen garments by implementing a Siamese network to learn the geometric similarities between garments and making predictions based on continuously perceiving the video frames of grasping garments with an "early-stop" strategy, where accuracy is at least 15% better than SOTA, and the pipeline requires less than 10 seconds.

This hypothesis has been validated in Chapter 4. The robot successfully predicted shapes and visually perceived weights of unseen garments by continuously perceiving video frames of garments. The results of Chapter 4 showed that the garment similarity network outperforms the CNN-LSTM network in Chapter 3 by 89.8%, indicating that learning from garment similarity maps is more efficient than the traditional classification approach.

The third hypothesis is:

- Predicting the bending stiffness parameters and area weights of real garments and fabrics can be achieved by learning physics similarities between simulated fabrics with a Siamese network, which outperforms SOTA by at least 30%;

This hypothesis has been validated in Chapter 5, where the proposed approach can successfully predict the area weights and bending stiffness parameters of seven fabrics and three garments by learning physics similarities between simulated fabrics. Chapter 5 also showed that the prediction of the physics properties of complex deformable objects, such as garments, can be achieved by learning the physics similarities between simple deformable objects, such as fabrics (with an overall error of 19.7% for wind speed predictions and 6.4% for area weight predictions).

The fourth hypothesis is:

- A robot can pick up crumpled garments from any location, recognise their *known con-*

figurations, and select pre-designed manipulation strategies based on recognised *known configurations*, where recognition accuracy is at least 80%.;

Chapter 6 has validated this hypothesis by constructing a robotic garment flattening pipeline. The results in Chapter 6 showed that a robot can successfully recognise garment *known configurations* based on their 3D maps. An offline pre-designed manipulation strategy was developed in order to flatten garments without online updates, boosting manipulation speed and improving the efficiency of manipulations with an average manipulation duration of 221.6 seconds.

The fifth hypothesis is

- A robot can flatten crumpled garments by picking them up from any location, predicting garment shapes, recognising the *known configurations* of garments based on predicted garment shapes, and selecting pre-designed manipulation strategies to flatten garments, where recognition accuracy is at least 90%, and flattening garments require less than 250 seconds with a success rate over 60%

Chapter 7 validate this hypothesis. In this chapter, a robot benefited from priorly predicting garments' geometric properties (shapes) for recognising garment *known configurations*. Recognition accuracy of *known configurations* increased from 89.0% in Chapter 6 to 92.0% in this chapter. The robot was thus more capable of distinguishing between sweaters and shirts compared to Chapter 6. The average manipulation time is 221.6 seconds, less than 250 seconds in the hypothesis. Chapter 7 finalises this thesis by introducing an effective robotic garment flattening pipeline by priorly predicting geometric properties (shapes) of garments, recognising garment *known configurations* based on predicted garment shapes and selecting pre-designed manipulation strategies to flatten garments.

8.3 Limitations and Future Work

Garments and other deformable object manipulation is one of the most challenging tasks in robotic manipulation. This thesis advanced the state of art by introducing an effective robotic garment flattening pipeline, a continuous-perception mechanism and a simulation-reality knowledge transfer paradigm to predict the physics properties of garments and fabrics. However, there are still limitations to this thesis. The following sections discuss the limitations in this thesis and provide potential solutions for future work.

8.3.1 Continuous-perception paradigm

This thesis develops two paradigms of continuous-perception mechanism in Chapters 3 and 4. However, both paradigms can only predict shapes and visually perceived weights of unseen garments if the shapes and visually perceived weights are one of five/three categories. In Chapter 4, a video frame of unseen garments can be labelled only when its encoded decision point is within one of the confidence intervals, which means decision points outside the confidence interval can not be labelled. The paradigms are limited to known shapes or visually perceived weights, while other shapes such as trousers, socks and scarfs cannot be predicted or wrongly predicted.

One possible direction for future work is to investigate continual learning [151]. The inclusion of all shapes and visually perceived weights is impractical for the continuous-perception mechanism. However, robots can learn unknown shapes from testing rather than training. Video frames from garments of unknown shapes or visually perceived weights will be mapped nearby on garment similarity maps. Robots can predict unknown shapes or visually perceived weights if a new confidence interval can be defined from newly formed clustering groups. Implementing continual learning in the continuous-perception mechanism also reduces training data size as the robot can learn from testing data. By following this approach, the robot can be used in more general settings: for example, garment laundry, where the shapes of garments are various and numerous.

8.3.2 Simulation-reality knowledge transfer paradigm

This thesis successfully predicts the physics properties of real garments and fabrics by learning physics similarities between simulated fabrics. But there is a limitation in that large search spaces of physics parameters can lead to prediction failures. This limitation causes the paradigm to be only applicable to certain materials. Garments and fabrics can be manufactured with various materials, while Chapter 5 only investigates denim, cotton and nylon.

In future work, the proposed physics similarity network (PhysSNet) should be implemented into the robotic garment flattening pipeline in Chapter 7. The robot can hold the garments, and an electric fan can wave them. Then their physics properties can be learned, and the learned physics properties can be used for facilitating recognising the *known configurations* of garments. Chapter 7 demonstrates the benefits of learning garments' geometric properties (shapes) in the robotic garment flattening pipeline. Learning the physics properties of garments can even further improve the performance of the pipeline.

Also, the physics properties can be used in a reinforcement or imitation learning approach to learn the manipulation skills of robots. Introducing the physics properties of garments can reduce action search space, improving reinforcement and imitation learning.

8.3.3 The proposed robotic garment flattening pipeline

Chapters 6 and 7 demonstrated an effective robotic garment flattening pipeline. However, a limitation in this pipeline is that the grippers of the robot can not handle the surface overlaps of garments properly. Garments were placed on the table with crumpled configurations, some were overlapped configurations. Overlaps cause the grippers to grasp both frontal and back layers of garments, which affects finding grasping points in later stages. Although the problem occasionally happened during the experiments, it caused manipulation failures. This limitation is because the proposed pipeline lacks more sophisticated manipulation skills for robots.

Overlaps cause problems; thus there are two proposed solutions. The first solution is detecting garment overlaps and refraining the robot from grasping garments from these overlaps. In future work, an overlap detection algorithm can be proposed to learn the features of overlaps from their depth maps or other information. Overlaps have a specific range of depth values, which should be easily detected from depth maps. Therefore, the robot firstly identifies overlaps with the proposed overlap detection algorithm and avoids grasping garments from the detected overlaps.

Also, the second solution is training the robot with reinforcement or imitation learning approaches to learn more skills for manipulating garments. Pore *et al.* [132] proposed learning robotic grasping skills from a behaviour-based reinforcement learning approach, which can be used in future work for robotic garment-manipulation skill learning.

Meanwhile, future work can focus on investigating the chaotic levels [152] of garments. Chapters 6 and 7 claims that the *known configurations* depend on the grasping points of garments rather than starting configurations. Therefore, further research can measure the chaotic levels of garments, which means research on whether garments have an infinite number of configurations (object states). If a garment is proven to have only a limited number of configurations, even without a need to be grasped from a table, a limited number of manipulation strategies can be designed and implemented to manipulate this garment, where these strategies do not need to update during manipulation. This future research can improve the efficiency and effectiveness of robotic garment manipulation pipelines.

Bibliography

[1] T. F. H. Runia, K. Gavrilyuk, C. G. M. Snoek, and A. W. M. Smeulders, "Cloth in the wind: A case study of physical measurement through simulation," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10 495–10 504.

[2] L. Sun, S. Rogers, G. Aragon-Camarasa, and J. P. Siebert, "Recognising the clothing categories from free-configuration using gaussian-process-based interactive perception," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 2464–2470.

[3] L. Martínez, J. R. del Solar, L. Sun, J. P. Siebert, and G. Aragon-Camarasa, "Continuous perception for deformable objects understanding," *Robotics and Autonomous Systems*, vol. 118, pp. 220 – 230, 2019.

[4] L. Sun, G. Aragon-Camarasa, S. Rogers, R. Stolkin, and J. P. Siebert, "Single-shot clothing category recognition in free-configurations with application to autonomous clothes sorting," *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 6699–6706, 2017.

[5] S. Yang, J. Liang, and M. C. Lin, "Learning-based cloth material recovery from video," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4383–4393.

[6] Y. Li, Y. Wang, Y. Yue, D. Xu, M. Case, S.-F. Chang, E. Grinspun, and P. K. Allen, "Model-driven feedforward prediction for manipulation of deformable objects," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 4, pp. 1621–1638, 2018.

[7] Y. Li, D. Xu, Y. Yue, Y. Wang, S.-F. Chang, E. Grinspun, and P. K. Allen, "Regrasping and unfolding of garments using predictive thin shell modeling," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 1382–1388.

[8] W. Yan, A. Vangipuram, P. Abbeel, and L. Pinto, “Learning predictive representations for deformable objects using contrastive estimation,” *arXiv preprint arXiv:2003.05436*, 2020.

[9] B. Balaguer and S. Carpin, “Combining imitation and reinforcement learning to fold deformable planar objects,” in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2011, pp. 1405–1412.

[10] M. Mansouri and F. Pecora, “Including qualitative spatial knowledge in the sense-plan-act loop,” in *International Conference on Constraint Processing, Doctoral Program.*, 2013.

[11] K. L. Bouman, B. Xiao, P. Battaglia, and W. T. Freeman, “Estimating the material properties of fabric from video,” in *Proceedings of the IEEE international conference on computer vision*, 2013, pp. 1984–1991.

[12] B. O. Community, *Blender - a 3D modelling and rendering package*, Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. [Online]. Available: http://www.blender.org

[13] J. K. Haas, “A history of the unity game engine,” *Diss. WORCESTER POLYTECHNIC INSTITUTE*, vol. 483, p. 484, 2014.

[14] K.-J. Bathe, “Finite element method,” *Wiley encyclopedia of computer science and engineering*, pp. 1–12, 2007.

[15] L. Twardon and H. Ritter, “Learning to put on a knit cap in a head-centric policy space,” *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 764–771, 2018.

[16] D. Chicco, “Siamese neural networks: An overview,” *Artificial neural networks*, pp. 73–94, 2021.

[17] L. Duan. and G. Aragon-Camarasa., “Continuous perception for classifying shapes and weights of garments for robotic vision applications,” in *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 4: VISAPP,*, 2022, pp. 348–355.

[18] L. Duan, L. Boyd, and G. Aragon-Camarasa, “Learning physics properties of fabrics and garments with a physics similarity neural network,” 2021. [Online]. Available: https://arxiv.org/abs/2112.10727

[19] L. Duan and G. Aragon-Camarasa, “A continuous robot vision approach for predicting shapes and visually perceived weights of garments,” *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7950–7957, 2022.

[20] L. Duan and G. Argon-Camarasa, "Recognising known configurations of garments for dual-arm robotic flattening," 2022. [Online]. Available: https://arxiv.org/abs/2205.00225

[21] L. Duan and G. Aragon-Camarasa, "A data-centric approach for dual-arm robotic garment flattening," *arXiv preprint arXiv:2208.13695*, 2022.

[22] M. Görner, R. Haschke, H. Ritter, and J. Zhang, "Moveit! task constructor for task-level motion planning," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 190–196.

[23] Y. Li, D. Xu, Y. Yue, Y. Wang, S.-F. Chang, E. Grinspun, and P. K. Allen, "Regrasping and unfolding of garments using predictive thin shell modeling," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 1382–1388.

[24] Y. Li, Y. Wang, Y. Yue, D. Xu, M. Case, S.-F. Chang, E. Grinspun, and P. K. Allen, "Model-driven feedforward prediction for manipulation of deformable objects," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 4, pp. 1621–1638, 2018.

[25] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, A. Y. Ng *et al.*, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.

[26] D. Coleman, I. Sucan, S. Chitta, and N. Correll, "Reducing the barrier to entry of complex robotic software: a moveit! case study," *arXiv preprint arXiv:1404.3785*, 2014.

[27] S. Kawabata and M. Niwa, "Fabric performance in clothing and clothing manufacture," *Journal of the Textile Institute*, vol. 80, no. 1, pp. 19–50, 1989.

[28] E. Miguel, D. Bradley, B. Thomaszewski, B. Bickel, W. Matusik, M. A. Otaduy, and S. Marschner, "Data-driven estimation of cloth simulation models," in *Computer Graphics Forum*, vol. 31. Wiley Online Library, 2012, pp. 519–528.

[29] D. Tanaka, S. Tsuda, and K. Yamazaki, "A learning method of dual-arm manipulation for cloth folding using physics simulator," in *2019 IEEE International Conference on Mechatronics and Automation (ICMA)*. IEEE, 2019, pp. 756–762.

[30] H. Wang, J. F. O'Brien, and R. Ramamoorthi, "Data-driven elastic models for cloth: modeling and measurement," *ACM transactions on graphics (TOG)*, vol. 30, no. 4, pp. 1–12, 2011.

[31] R. Narain, A. Samii, T. Pfaff, and J. O'Brien, "Arcsim: Adaptive refining and coarsening simulator," *University of California–Berkley, Berkley, CA, accessed Oct*, vol. 1, p. 2016, 2014.

[32] S. Bell, P. Upchurch, N. Snavely, and K. Bala, “Material recognition in the wild with the materials in context database,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3479–3487.

[33] A. Davis, K. L. Bouman, J. G. Chen, M. Rubinstein, F. Durand, and W. T. Freeman, “Visual vibrometry: Estimating material properties from small motion in video,” in *Proceedings of the ieee conference on computer vision and pattern recognition*, 2015, pp. 5335–5343.

[34] V. E. Arriola-Rios and J. L. Wyatt, “A multimodal model of object deformation under robotic pushing,” *IEEE Transactions on Cognitive and Developmental Systems*, vol. 9, no. 2, pp. 153–169, 2017.

[35] P. Krähenbühl and V. Koltun, “Parameter learning and convergent inference for dense random fields,” in *International Conference on Machine Learning*. PMLR, 2013, pp. 513–521.

[36] A. Sengupta, R. Lagneau, A. Krupa, E. Marchand, and M. Marchal, “Simultaneous tracking and elasticity parameter estimation of deformable objects,” in *IEEE Int. Conf. on Robotics and Automation, ICRA’20*, 2020.

[37] S. Yang, V. Jojic, J. Lian, R. Chen, H. Zhu, and M. C. Lin, “Classification of prostate cancer grades and t-stages based on tissue elasticity using medical image analysis,” in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2016, pp. 627–635.

[38] R. W. Fleming, R. O. Dror, and E. H. Adelson, “Real-world illumination and the perception of surface reflectance properties,” *Journal of vision*, vol. 3, no. 5, pp. 3–3, 2003.

[39] Y. Li, J. Wu, R. Tedrake, J. B. Tenenbaum, and A. Torralba, “Learning particle dynamics for manipulating rigid bodies, deformable objects, and fluids,” *arXiv preprint arXiv:1810.01566*, 2018.

[40] Y. Li, T. Lin, K. Yi, D. Bear, D. L. K. Yamins, J. Wu, J. B. Tenenbaum, and A. Torralba, “Visual grounding of learned physical models,” 2020.

[41] K. S. Bhat, C. D. Twigg, J. K. Hodgins, P. K. Khosla, Z. Popović, and S. M. Seitz, “Estimating cloth simulation parameters from video,” in *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*. Eurographics Association, 2003, pp. 37–51.

[42] Y.-X. Ho, M. S. Landy, and L. T. Maloney, “How direction of illumination affects visually perceived surface roughness,” *Journal of vision*, vol. 6, no. 5, pp. 8–8, 2006.

[43] B. Tawbe and A. Crétu, “Acquisition and neural network prediction of 3d deformable object shape using a kinect and a force-torque sensor †,” *Sensors (Basel, Switzerland)*, vol. 17, 2017.

[44] Z. Wang, S. Rosa, B. Yang, S. Wang, N. Trigoni, and A. Markham, “3d-physnet: Learning the intuitive physics of non-rigid object deformations,” *arXiv preprint arXiv:1805.00328*, 2018.

[45] A. Doumanoglou, A. Kargakos, T.-K. Kim, and S. Malassiotis, “Autonomous active recognition and unfolding of clothes using random decision forests and probabilistic planning,” in *2014 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2014, pp. 987–993.

[46] C. Chi and S. Song, “Garmentnets: Category-level pose estimation for garments via canonical space shape completion,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 3324–3333.

[47] B. Willimon, S. Birchfield, and I. Walker, “Classification of clothing using interactive perception,” in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 1862–1868.

[48] A. Ramisa, G. Alenya, F. Moreno-Noguer, and C. Torras, “Learning rgb-d descriptors of garment parts for informed robot grasping,” *Engineering Applications of Artificial Intelligence*, vol. 35, pp. 246–258, 2014.

[49] B. Willimon, I. Walker, and S. Birchfield, “A new approach to clothing classification using mid-level layers,” in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 4271–4278.

[50] Y. Li, C.-F. Chen, and P. K. Allen, “Recognition of deformable object category and pose,” in *2014 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2014, pp. 5558–5564.

[51] I. Mariolis, G. Peleka, A. Kargakos, and S. Malassiotis, “Pose and category recognition of highly deformable objects using deep learning,” in *2015 International Conference on Advanced Robotics (ICAR)*. IEEE, 2015, pp. 655–662.

[52] A. Gabas, E. Corona, G. Alenyà, and C. Torras, “Robot-aided cloth classification using depth information and cnns,” in *International Conference on Articulated Motion and Deformable Objects*. Springer, 2016, pp. 16–23.

[53] E. Corona, G. Alenya, A. Gabas, and C. Torras, “Active garment recognition and target grasping point detection using deep learning,” *Pattern Recognition*, vol. 74, pp. 629–641, 2018.

[54] C. Kampouris, I. Mariolis, G. Peleka, E. Skartados, A. Kargakos, D. Triantafyllou, and S. Malassiotis, "Multi-sensorial and explorative recognition of garments and their material properties in unconstrained environment," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 1656–1663.

[55] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, "Deepfashion: Powering robust clothes recognition and retrieval with rich annotations," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1096–1104.

[56] J. Huang, R. S. Feris, Q. Chen, and S. Yan, "Cross-domain image retrieval with a dual attribute-aware ranking network," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1062–1070.

[57] W. Wang, Y. Xu, J. Shen, and S.-C. Zhu, "Attentive fashion grammar network for fashion landmark detection and clothing category classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4271–4280.

[58] J. Liu and H. Lu, "Deep fashion analysis with feature map upsampling and landmark-driven attention," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018, pp. 0–0.

[59] T. Ziegler, J. Butepage, M. C. Welle, A. Varava, T. Novkovic, and D. Kragic, "Fashion landmark detection and category classification for robotics," in *2020 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*. IEEE, 2020, pp. 81–88.

[60] O. Gustavsson, T. Ziegler, M. C. Welle, J. Bütepage, A. Varava, and D. Kragic, "Cloth manipulation based on category classification and landmark detection," *International Journal of Advanced Robotic Systems*, vol. 19, no. 4, p. 17298806221110445, 2022.

[61] L. Wagner, D. Krejcová, and V. Smutnỳ, "Ctu color and depth image dataset of spread garments," *Center for Machine Perception, Czech Technical University, Tech. Rep. CTU-CMP-2013-25*, 2013.

[62] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[63] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[64] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[65] J.-H. Kim, J. Jun, and B.-T. Zhang, "Bilinear attention networks," *Advances in neural information processing systems*, vol. 31, 2018.

[66] H. Lin, F. Guo, F. Wang, and Y.-B. Jia, "Picking up a soft 3d object by "feeling" the grip," *The International Journal of Robotics Research*, vol. 34, no. 11, pp. 1361–1384, 2015.

[67] T. Cui, J. Xiao, and A. Song, "Simulation of grasping deformable objects with a virtual human hand," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2008, pp. 3965–3970.

[68] Y.-B. Jia, F. Guo, and H. Lin, "Grasping deformable planar objects: Squeeze, stick/slip analysis, and energy-based optimalities," *The International Journal of Robotics Research*, vol. 33, no. 6, pp. 866–897, 2014.

[69] I. Farmaga, P. Shmigelskyi, P. Spiewak, and L. Ciupinski, "Evaluation of computational complexity of finite element analysis," in *2011 11th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM)*. IEEE, 2011, pp. 213–214.

[70] J. Tarrier, A. Harland, R. Jones, T. Lucas, and D. Price, "Applying finite element analysis to compression garment development," *Procedia Engineering*, vol. 2, no. 2, pp. 3349–3354, 2010.

[71] F. Strazzeri and C. Torras, "Topological representation of cloth state for robot manipulation: Deriving the configuration space of a rectangular cloth," *Autonomous Robots*, vol. 45, no. 5, pp. 737–754, 2021.

[72] E. Nabil, B. Belhassen-Chedli, and G. Grigore, "Soft material modeling for robotic task formulation and control in the muscle separation process," *Robotics and Computer-Integrated Manufacturing*, vol. 32, pp. 37–53, 2015.

[73] L. Zaidi, J. A. Corrales, B. C. Bouzgarrou, Y. Mezouar, and L. Sabourin, "Model-based strategy for grasping 3d deformable objects using a multi-fingered robotic hand," *Robotics and Autonomous Systems*, vol. 95, pp. 196–206, 2017.

[74] A. Simeonov, Y. Du, B. Kim, F. R. Hogan, J. Tenenbaum, P. Agrawal, and A. Rodriguez, "A long horizon planning framework for manipulating rigid pointcloud objects," 2020.

[75] J. Schulman, A. Lee, J. Ho, and P. Abbeel, "Tracking deformable objects with point clouds," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 1130–1137.

[76] X. Lin, Y. Wang, Z. Huang, and D. Held, "Learning visible connectivity dynamics for cloth smoothing," in *Conference on Robot Learning*. PMLR, 2022, pp. 256–266.

[77] T. M. Martinetz, S. G. Berkovich, and K. J. Schulten, "'neural-gas' network for vector quantization and its application to time-series prediction," *IEEE transactions on neural networks*, vol. 4, no. 4, pp. 558–569, 1993.

[78] M. Moletta, M. C. Welle, A. Kravchenko, A. Varava, and D. Kragic, "Representing clothing items for robotics tasks," in *KTH Royal Institute of Technology*, 2022.

[79] R. Antonova, A. Varava, P. Shi, J. F. Carvalho, and D. Kragic, "Sequential topological representations for predictive models of deformable objects," in *Learning for Dynamics and Control*. PMLR, 2021, pp. 348–360.

[80] S. Miller, J. Van Den Berg, M. Fritz, T. Darrell, K. Goldberg, and P. Abbeel, "A geometric approach to robotic laundry folding," *The International Journal of Robotics Research*, vol. 31, no. 2, pp. 249–267, 2012.

[81] P. Güler, A. Pieropan, M. Ishikawa, and D. Kragic, "Estimating deformability of objects using meshless shape matching," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 5941–5948.

[82] W. Yu, A. Kapusta, J. Tan, C. C. Kemp, G. Turk, and C. K. Liu, "Haptic simulation for robot-assisted dressing," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 6044–6051.

[83] D. Tanaka, S. Arnold, and K. Yamazaki, "Emd net: An encode–manipulate–decode network for cloth manipulation," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1771–1778, 2018.

[84] E. Pignat and S. Calinon, "Learning adaptive dressing assistance from human demonstration," *Robotics and Autonomous Systems*, vol. 93, pp. 61–75, 2017.

[85] S. H. Huang, J. Pan, G. Mulcaire, and P. Abbeel, "Leveraging appearance priors in non-rigid registration, with application to manipulation of deformable objects," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 878–885.

[86] D. Seita, A. Ganapathi, R. Hoque, M. Hwang, E. Cen, A. K. Tanwani, A. Balakrishna, B. Thananjeyan, J. Ichnowski, N. Jamali *et al.*, "Deep imitation learning of sequential fabric smoothing from an algorithmic supervisor," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 9651–9658.

[87] S. Ross, G. Gordon, and D. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011, pp. 627–635.

[88] R. Lee, D. Ward, A. Cosgun, V. Dasagi, P. Corke, and J. Leitner, “Learning arbitrary-goal fabric folding with one hour of real robot experience,” *arXiv preprint arXiv:2010.03209*, 2020.

[89] D. Quillen, E. Jang, O. Nachum, C. Finn, J. Ibarz, and S. Levine, “Deep reinforcement learning for vision-based robotic grasping: A simulated comparative evaluation of off-policy methods,” in *2018 IEEE International Conference on Robotics and Automation (ICRA).* IEEE, 2018, pp. 6284–6291.

[90] H. Van Hasselt, A. Guez, and D. Silver, “Deep reinforcement learning with double q-learning,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 30, no. 1, 2016.

[91] A. X. Lee, H. Lu, A. Gupta, S. Levine, and P. Abbeel, “Learning force-based manipulation of deformable objects from multiple demonstrations,” in *2015 IEEE International Conference on Robotics and Automation (ICRA).* IEEE, 2015, pp. 177–184.

[92] J. Matas, S. James, and A. J. Davison, “Sim-to-real reinforcement learning for deformable object manipulation,” in *Conference on Robot Learning.* PMLR, 2018, pp. 734–743.

[93] Y. Wu, W. Yan, T. Kurutach, L. Pinto, and P. Abbeel, “Learning to manipulate deformable objects without demonstrations,” *arXiv preprint arXiv:1910.13439*, 2019.

[94] T. Matsubara, D. Shinohara, and M. Kidode, “Reinforcement learning of a motor skill for wearing a t-shirt using topology coordinates,” *Advanced Robotics*, vol. 27, no. 7, pp. 513–524, 2013.

[95] A. Colomé, A. Planells, and C. Torras, “A friction-model-based framework for reinforcement learning of robotic tasks in non-rigid environments,” in *2015 IEEE international conference on robotics and automation (ICRA).* IEEE, 2015, pp. 5649–5654.

[96] Y. Gao, H. J. Chang, and Y. Demiris, “Iterative path optimisation for personalised dressing assistance using vision and force information,” in *2016 IEEE/RSJ international conference on intelligent robots and systems (IROS).* IEEE, 2016, pp. 4398–4403.

[97] Y. Tsurumine, Y. Cui, E. Uchibe, and T. Matsubara, “Deep reinforcement learning with smooth policy update: Application to robotic cloth manipulation,” *Robotics and Autonomous Systems*, vol. 112, pp. 72–83, 2019.

[98] R. Jangir, G. Alenya, and C. Torras, “Dynamic cloth manipulation with deep reinforcement learning,” in *2020 IEEE International Conference on Robotics and Automation (ICRA).* IEEE, 2020, pp. 4630–4636.

[99] D. McConachie and D. Berenson, “Estimating model utility for deformable object manipulation using multiarmed bandit methods,” *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 3, pp. 967–979, 2018.

[100] F. Ebert, C. Finn, S. Dasari, A. Xie, A. Lee, and S. Levine, “Visual foresight: Model-based deep reinforcement learning for vision-based robotic control,” *arXiv preprint arXiv:1812.00568*, 2018.

[101] J. Stria, V. Petrík, and V. Hlaváč, “Model-free approach to garments unfolding based on detection of folded layers,” in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 3274–3280.

[102] D. Estevez, J. G. Victores, R. Fernandez-Fernandez, and C. Balaguer, “Enabling garment-agnostic laundry tasks for a robot household companion,” *Robotics and Autonomous Systems*, vol. 123, p. 103330, 2020.

[103] D. Triantafyllou and N. A. Aspragathos, “A vision system for the unfolding of highly non-rigid objects on a table by one manipulator,” in *International Conference on Intelligent Robotics and Applications*. Springer, 2011, pp. 509–519.

[104] D. Triantafyllou, I. Mariolis, A. Kargakos, S. Malassiotis, and N. Aspragathos, “A geometric approach to robotic unfolding of garments,” *Robotics and Autonomous Systems*, vol. 75, pp. 233–243, 2016.

[105] D. Triantafyllou, P. Koustoumpardis, and N. Aspragathos, “Type independent hierarchical analysis for the recognition of folded garments’ configuration,” *Intelligent Service Robotics*, vol. 14, no. 3, pp. 427–444, 2021.

[106] D. Triantafyllou and N. Aspragathos, “Garment type agnostic robotic unfolding of garments from random configuration,” in *International Conference on Robotics in Alpe-Adria Danube Region*. Springer, 2022, pp. 487–495.

[107] F. Osawa, H. Seki, and Y. Kamiya, “Unfolding of massive laundry and classification types by dual manipulator,” *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 11, no. 5, pp. 457–463, 2007.

[108] M. Cusumano-Towner, A. Singh, S. Miller, J. F. O’Brien, and P. Abbeel, “Bringing clothing into desired configurations with limited perception,” in *2011 IEEE international conference on robotics and automation*. IEEE, 2011, pp. 3893–3900.

[109] J. Maitin-Shepard, M. Cusumano-Towner, J. Lei, and P. Abbeel, “Cloth grasp point detection based on multiple-view geometric cues with application to robotic towel folding,” in *2010 IEEE International Conference on Robotics and Automation*. IEEE, 2010, pp. 2308–2315.

[110] B. Willimon, S. Birchfield, and I. Walker, “Model for unfolding laundry using interactive perception,” in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2011, pp. 4871–4876.

[111] H. Yuba, S. Arnold, and K. Yamazaki, “Unfolding of a rectangular cloth from unarranged starting shapes by a dual-armed robot with a mechanism for managing recognition error and uncertainty,” *Advanced Robotics*, vol. 31, no. 10, pp. 544–556, 2017.

[112] H. Ha and S. Song, “Flingbot: The unreasonable effectiveness of dynamic manipulation for cloth unfolding,” in *Conference on Robot Learning*. PMLR, 2022, pp. 24–33.

[113] J. Qian, T. Weng, L. Zhang, B. Okorn, and D. Held, “Cloth region segmentation for robust grasp selection,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 9553–9560.

[114] D. Seita, N. Jamali, M. Laskey, A. K. Tanwani, R. Berenstein, P. Baskaran, S. Iba, J. Canny, and K. Goldberg, “Deep transfer learning of pick points on fabric for robot bed-making,” in *The International Symposium of Robotics Research*. Springer, 2019, pp. 275–290.

[115] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[116] Y. Kita, F. Saito, and N. Kita, “A deformable model driven visual method for handling clothes,” in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA’04. 2004*, vol. 4. IEEE, 2004, pp. 3889–3895.

[117] J. Bohg, K. Hausman, B. Sankaran, O. Brock, D. Kragic, S. Schaal, and G. S. Sukhatme, “Interactive perception: Leveraging action in perception and perception in action,” *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1273–1291, 2017.

[118] P. Jiménez and C. Torras, “Perception of cloth in assistive robotic manipulation tasks,” *Natural Computing*, pp. 1–23, 2020.

[119] A. Ganapathi, P. Sundaresan, B. Thananjeyan, A. Balakrishna, D. Seita, J. Grannen, M. Hwang, R. Hoque, J. E. Gonzalez, N. Jamali, K. Yamane, S. Iba, and K. Goldberg, “Learning to smooth and fold real fabric using dense object descriptors trained on synthetic color images,” 2020.

[120] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Advances in neural information processing systems*, vol. 25, 2012.

[121] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[122] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[123] B. Willimon, S. Birchfield, and I. Walker, "Classification of clothing using interactive perception," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 1862–1868.

[124] L. Sun, S. Rogers, G. Aragon-Camarasa, and J. P. Siebert, "Recognising the clothing categories from free-configuration using gaussian-process-based interactive perception," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 2464–2470.

[125] L. Sun, G. Aragon-Camarasa, S. Rogers, and J. P. Siebert, "Accurate garment surface analysis using an active stereo robot head with application to dual-arm flattening," in *2015 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2015, pp. 185–192.

[126] A. Doumanoglou, J. Stria, G. Peleka, I. Mariolis, V. Petrik, A. Kargakos, L. Wagner, V. Hlaváč, T.-K. Kim, and S. Malassiotis, "Folding clothes autonomously: A complete pipeline," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1461–1478, 2016.

[127] E. Hoffer and N. Ailon, "Deep metric learning using triplet network," in *International workshop on similarity-based pattern recognition*. Springer, 2015, pp. 84–92.

[128] M. Rosenblatt, "Remarks on Some Nonparametric Estimates of a Density Function," *The Annals of Mathematical Statistics*, vol. 27, no. 3, pp. 832 – 837, 1956.

[129] H. M. Bui, M. Lech, E. Cheng, K. Neville, and I. S. Burnett, "Using grayscale images for object recognition with convolutional-recursive neural network," in *2016 IEEE Sixth International Conference on Communications and Electronics (ICCE)*. IEEE, 2016, pp. 321–325.

[130] M. B. Shaikh and D. Chai, "Rgb-d data-based action recognition: a review," *Sensors*, vol. 21, no. 12, p. 4246, 2021.

[131] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, "Progressive neural networks," *arXiv preprint arXiv:1606.04671*, 2016.

[132] A. Pore and G. Aragon-Camarasa, “On simple reactive neural networks for behaviour-based reinforcement learning,” in *2020 IEEE International Conference on Robotics and Automation (ICRA).* IEEE, 2020, pp. 7477–7483.

[133] L. Sun, G. Aragon-Camarasa, S. Rogers, and J. P. Siebert, “Accurate garment surface analysis using an active stereo robot head with application to dual-arm flattening,” in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 185–192.

[134] L. Duan, L. Boyd, and G. Aragon-Camarasa, “Learning physics properties of fabrics and garments with a physics similarity neural network,” 2021. [Online]. Available: https://arxiv.org/abs/2112.10727

[135] R. Hess, *Blender Foundations: The Essential Guide to Learning Blender 2.6.* Focal Press, 2010.

[136] E. Coumans and Y. Bai, “Pybullet, a python module for physics simulation for games, robotics and machine learning,” http://pybullet.org, 2021.

[137] Y. Yamakawa, A. Namiki, and M. Ishikawa, “Simple model and deformation control of a flexible rope using constant, high-speed motion of a robot arm,” in *2012 IEEE International Conference on Robotics and Automation.* IEEE, 2012, pp. 2249–2254.

[138] T. H. Courtney, *Mechanical behavior of materials.* Waveland Press, 2005.

[139] A. Sengupta, R. Lagneau, A. Krupa, E. Marchand, and M. Marchal, “Simultaneous tracking and elasticity parameter estimation of deformable objects,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 10 038–10 044.

[140] R. Narain, A. Samii, and J. F. O’Brien, “Adaptive anisotropic remeshing for cloth simulation,” *ACM Trans. Graph.*, vol. 31, no. 6, Nov. 2012. [Online]. Available: https://doi.org/10.1145/2366145.2366171

[141] G. Koch, R. Zemel, and R. Salakhutdinov, “Siamese neural networks for one-shot image recognition,” in *ICML deep learning workshop*, vol. 2. Lille, 2015.

[142] B. Ghojogh, M. Sikaroudi, S. Shafiei, H. R. Tizhoosh, F. Karray, and M. Crowley, “Fisher discriminant triplet and contrastive losses for training siamese networks,” in *2020 international joint conference on neural networks (IJCNN).* IEEE, 2020, pp. 1–7.

[143] M. Balandat, B. Karrer, D. Jiang, S. Daulton, B. Letham, A. G. Wilson, and E. Bakshy, “Botorch: A framework for efficient monte-carlo bayesian optimization,” *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[144] Y. Li, Y. Yue, D. Xu, E. Grinspun, and P. K. Allen, “Folding deformable objects using predictive simulation and trajectory optimization,” in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 6000–6006.

[145] D. Agrawal, M. Choy, H. Va Leong, and A. K. Singh, “Maya: A simulation platform for distributed shared memories,” in *Proceedings of the Eighth Workshop on Parallel and Distributed Simulation*, ser. PADS ’94. New York, NY, USA: Association for Computing Machinery, 1994, p. 151–155. [Online]. Available: https://doi.org/10.1145/182478.182583

[146] J. Schulman, J. Ho, C. Lee, and P. Abbeel, “Learning from demonstrations through the use of non-rigid registration,” in *Robotics Research*. Springer, 2016, pp. 339–354.

[147] L. Duan and G. Aragon-Camarasa, “Garnet: A continuous robot vision approach for predicting shapes and visually perceived weights of garments,” 2022. [Online]. Available: https://arxiv.org/abs/2109.07831

[148] B. Tawbe and A.-M. Cretu, “Acquisition and neural network prediction of 3d deformable object shape using a kinect and a force-torque sensor,” *Sensors*, vol. 17, no. 5, p. 1083, 2017.

[149] A. Khan, G. Aragon-Camarasa, L. Sun, and J. P. Siebert, “On the calibration of active binocular and rgbd vision systems for dual-arm robots,” in *2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2016, pp. 1960–1965.

[150] J. Maitin-Shepard, M. Cusumano-Towner, J. Lei, and P. Abbeel, “Cloth grasp point detection based on multiple-view geometric cues with application to robotic towel folding,” in *2010 IEEE International Conference on Robotics and Automation*, 2010, pp. 2308–2315.

[151] M. De Lange, R. Aljundi, M. Masana, S. Parisot, X. Jia, A. Leonardis, G. Slabaugh, and T. Tuytelaars, “A continual learning survey: Defying forgetting in classification tasks,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 7, pp. 3366–3385, 2021.

[152] C. Oestreicher, “A history of chaos theory,” *Dialogues in clinical neuroscience*, 2022.