TECHNISCHE UNIVERSITÄT MÜNCHEN
Lehrstuhl für Echtzeitsysteme und Robotik

DYNAMIC SCENE UNDERSTANDING AND RECOGNITION FROM PROPERTIES OF VISUAL DATA STRUCTURES

Wei Wang

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender	Univ.-Prof. Dr.-Ing. Alin Albu-Schäffer
Prüfer der Dissertation:	1. Univ.-Prof. Dr.-Ing. Darius Burschka
	2. Prof. Gregory D. Hager, Ph.D. Johns Hopkins University, Baltimore / USA

Die Dissertation wurde am 29.09.2014 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 26.01.2015 angenommen.

"If opportunity doesn't knock, build a door."

— Milton Berle

To my Lili, Yicheng and Family

ABSTRACT

Scene understanding and recognition in dynamic environments is a primary goal of cognitive perception systems. "See", "analyze" and "understand" are three keywords best describing the ultimate objectives of the perception system. By utilizing hybrid visual sensors and fulfilling the requirements of different application scenarios, the dynamic environment is represented as multiple data structures at various layers of abstraction. There has been significant interest in recent years in effectively and efficiently extracting useful information from the analysis of the properties of such visual data structures. In this thesis, we present multiple contributions to scene understanding and recognition in the areas of both texture and textureless object recognition as well as pose estimation, dense/deformable motion extraction for dynamic scene, modeling of articulated object, and real-time human motion estimation.

We propose an object descriptor in the form of a viewpoint oriented color-shape histogram for object recognition and pose estimation, which combines the object's color and shape features. The descriptor is efficiently used in a real-time textured/textureless object recognition and 6D pose estimation system. We validate our approach through a large number of experiments, including daily complex scenarios and object localization in a large-scale semantic map. Secondly, a hierarchical MRF optimization method is designed for dense and deformable motion extraction from dynamic scenes. It consists of two layers, named the segmentation and the correspondence layer respectively. The dynamic foreground data is first segmented and then used to estimate motion by employing point correspondences. A new surface descriptor aptly titled the "deformable color and shape histogram" is proposed and a dataset of dynamic scenes is built for benchmarking purposes, which involves different motion patterns and surface properties. We propose a new articulated object modeling method by integrating visual and manipulation information for articulated object recognition and manipulation. Depth-based skeletonization is realized to extract the skeleton from visual observations of different configurations. The kinematic joints are characterized and localized. Robot manipulation information is gained by kinesthetic learning in object joint state space. Following modeling, manipulation of the object is realized by first identifying the current object joint states and generalizing the learned force to accomplish the new task. Lastly, we propose a real-time 3D human body motion estimation based on three-layer laser scans. The relevant scanned points represent human body contour information and are subtracted from the image as part of feature extraction. In order to avoid situations of unsuccessful segmentation, a new iterative template

matching algorithm for clustering is proposed. The positions of human joints in 3D space are retrieved by associating the extracted features with a pre-defined articulated human body model while simultaneously estimating accurate human body motion in real time.

ZUSAMMENFASSUNG

Verständnis und Erkennung von Szenen in dynamischen Umgebung ist ein Hauptziel von kognitiven Wahrnehmungssystemen. Das "Sehen", die "Analyse" und das "Verständnis" sind drei Schlagwörter, die die Ziele vom Wahrnehmungssystem am Besten beschreiben. Unter Verwendung von hybriden Sensoren und unter Berücksichtigung der Anforderungen von diversen Anwendungsszenarien werden dynamische Umgebungen als verschiedene Datenstrukturen in unterschiedlichen Abstraktionsschichten realisiert. Es gibt aktuell signifikantes Interesse an der effektiven und effizienten Gewinnung von Informationen aus der Analyse der Eigenschaften solcher visuellen Strukturen. In dieser Arbeit werden Beiträge zum Verständnis und zur Erkennung von Szenen vorgestellt, die sowohl im Bereich der textur-und texturlosen Objekterkennung als auch zur Schätzung der Pose, der Bestimmung von dichter Bewegungsinformation für deformierbare Objekte in dynamischen Szenen, der Modellierung von beweglichen Objekten und der Bewegungsschätzung von Menschen in Echtzeit angesiedelt sind.

Wir schlagen einen Objektdeskriptor vor, der mithilfe eines betrachtungsabhängigen Farb-und Formhistogramms für Objekterkennung und Positionsbestimmung genutzt werden kann und welcher die Farb-und Formmerkmale eines Objekts kombiniert. Der Deskriptor wird effizient in einem System sowohl für die Erkennung von texturlosen und texturierten Objekten in Echtzeit als auch für die 6D Positionsbestimmung eingesetzt. Das Vorgehen wurde durch eine Vielzahl an Experimenten validiert, die alltägliche komplexe Szenarien nachbilden und die Lagebestimmung von Objekten in einer komplexen semantischen Karte testen. Zweitens wurde ein MRF-Optimierungsverfahren für dichte und deformierbare Bewegungen in dynamischen Szenen entworfen. Das Verfahren besteht aus zwei Schichten, die als Segmentierungs- und Korrespondenzschicht bezeichnet werden. Die dynamische Vordergrundinformationen werden zunächst segmentiert und im darauf folgenden Schritt für die Bewegungsschätzung mittels Punktkorrespondenzen verwendet. Ein neuer Oberflächendeskriptor wird vorgestellt, der als "deformierbares Farb- und Formhistogramm" bezeichnet wird. Zusätzlich wurde ein neuer Datensatz aus dynamischen Szenen zusammengestellt, der diverse Bewegungsmuster und Oberflächeneigenschaften beinhaltet, um eine Bewertung des Verfahrens zu ermöglichen. Außerdem wird eine neue Methode zur Modellierung von artikulierten Objekten vorgeschlagen, welche visuelle und manipulationsrelevante Informationen zur Erkennung und Manipulation von artikulierten Objekten integriert. Tiefenbasierte Skelettierung wird zur Extraktion des Skeletts aus visuellen Beobachtungen verschiedener Konfigurationen eingesetzt. Kinematische

Gelenke werden charakterisiert und lokalisiert. Information zur Manipulation mit einem Roboter wird durch Lernen der Kinästhetik im Zustandsraum der Objektgelenke vermittelt. Nach der Modellierung wird die Manipulation des Objektes realisiert, indem zuerst der aktuelle Zustand der Objektgelenke identifiziert wird und dann die gelernten Kräfte generalisiert werden, um neue Aufgaben zu erfüllen. Zum Schluss wird eine echtzeitfähige 3D Bewegungsschätzung von Menschen mithilfe von Laserscans mit drei Ebenen vorgestellt. Die erfassten Punkte stellen Konturinformationen des menschlichen Körpers dar und werden im Rahmen der Merkmalsextraktion vom Bild abgezogen. Damit erfolglose Segmentierungen vermieden werden, wird einen neuer iterativer Template-Matching Algorithmus für das Clustering vorgestellt. Die Positionen der Körpergelenke im 3D Raum werden durch Zuordnung der extrahierten Merkmale zum einem vordefinierten artikulierten menschlichen Modell bestimmt, gleichzeitig wird die Körperbewegung in Echtzeit mit hoher Genauigkeit geschätzt.

ACKNOWLEDGMENTS

First and foremost, I would like to sincere thank my supervisor Prof. Dr.-Ing. Darius Burschka to provide me an opportunity to conduct my research in his prestigious group. He gave me research guidance, professional expertise and comprehensive support. His encouragements provided me the concentration and freedom in pursuing my own ideas. Many thanks, my "Doctor Father".

Next, I would like to express my gratitude to all of the colleagues in Machine Vision and Perception Group (MVP) for their kind support. A special thanks to Dr. Konstantinos Dalamagkidis, Dr. Susanne Petsch, Dr. Michael Jäntsch, Dr. Steffen Wittmeier, Artashes Mkhitaryan, Philipp Heise, Brian Jensen, Rafael Hostettler, Benito Clemente Diaz Nava, Aurelien Bustin, Shufang Liu and Terresa Dominguez Rincón, for research discussions and enjoyable interactions. I also want to thank my co-authors for the insightful discussions, fruitful collaborations, and late-night paper writing sessions. Moreover, I want to thank my previous students and Murola people. I enjoyed every minute working with them.

Last but not least, my deepest gratitude goes to my family for their continuing support. And I also own great thanks to my wife Lili, who has been always standing by me and giving powerful encouragements during all these years. Also thanks to my daughter Yicheng, who brings a colorful life to me. Without their love and support, I cannot get these achievements and reach this step.

CONTENTS

1 INTRODUCTION

Scene understanding and recognition in dynamic environments is a primary goal of cognitive perception systems. A perception system may be defined as a system that retrieves, then organizes and finally interprets sensory information in order to represent and understand the environment. To retrieve sensory information, hybrid visual sensors, have been widely employed in lots of applications, provide significant advantages due to the richer information they provide. The collected information then could be typically organized in multiple data structures at various layers of abstraction. This data as well as the properties and relationships between the data structures can be used to extract useful information, which is needed in the last step (interpret) to understand the environment. A successful and robust perception system of dynamic scene understanding and recognition is therefore encouraged to be developed in massive research and industrial areas. For instance, in the application of robotics, the system brings effective abstractions of entire environments for autonomous robots, so that being able to adapt their actuators for different applications. With this capability, the robot can be applied as a real autonomous assistant for human beings in daily life, without pre-programmed processes.

In the last decades, a major research question as the topic of this thesis is how to effectively and efficiently extract the information from these data structures to fulfill the requirements of different application scenarios. The latter include topics such as object recognition as well as pose estimation, dense/deformable motion extraction in dynamic scenes, modeling of articulated objects, and real-time human motion estimation. This chapter introduces the background of scene understanding and recognition from visual data structures in dynamic environments. It then presents the research objectives, challenges and contributions of this thesis which are followed by an overview of its structure.

1.1 BACKGROUND

A scene understanding and recognition system comprises many low-, mid- and high-level visual computations such as sensing, construction and analysis. Due to their great importance, perception systems have attracted the great interests of vision researchers as well as researchers from the brain and neural science, psychology, cognitive science and computer science domains. Perception systems have been used in a number of

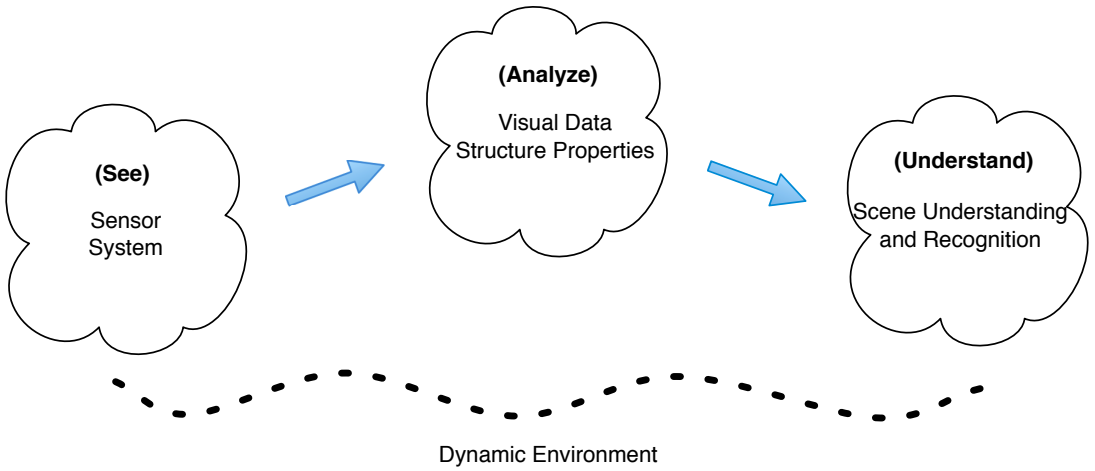


Figure 1.1: Perception system for scene understanding and recognition, which contains sensor system, analysis for visual structure properties and cognition process.

different applications, such as visual surveillance (object tracking and behavior analysis) [9, 13, 14, 24, 182], object exploration (recognition and pose estimation) [33, 51, 107, 122, 115], self-localization (autonomous robot exploration) [3, 43, 101, 113, 117] and scene analysis (dynamic motion extraction and prediction) [86, 97, 102, 123, 179]. In general, a perception system, as shown in Figure 1.1, consists of three main components as follows:

- "see", i.e. a sensor system to collect scene data;
- "analyze", i.e. the organization, analysis and extraction of useful information;
- "understand", an application-dependent system that "makes sense" out of the data.

There are normally two ways to represent a dynamic environment utilizing the scene data obtained from a sensor system. In the first case, static sensors are utilized and the movements of active agents provide the dynamic information. In the second case, the sensors are moving and the scene data are dynamic even if there is no moving object in the scene. For different working environments and applications, different sensors are developed and utilized to collect visual data of the dynamic scene. These sensors include RGB cameras, stereo cameras, lasers, time-of-flight cameras and RBG-D cameras such as the Microsoft Kinect among others. Networks of multiple sensors can be used to collect different types of data such as intensity, color and depth information. The raw sensor data can then be organized into different data structures for future analysis [133].

The aforementioned data structures can be categorized into "iconic images", "segmented images", "geometric representations" and "relational models". The "iconic images" category consists of images containing original data like intensity, brightness or

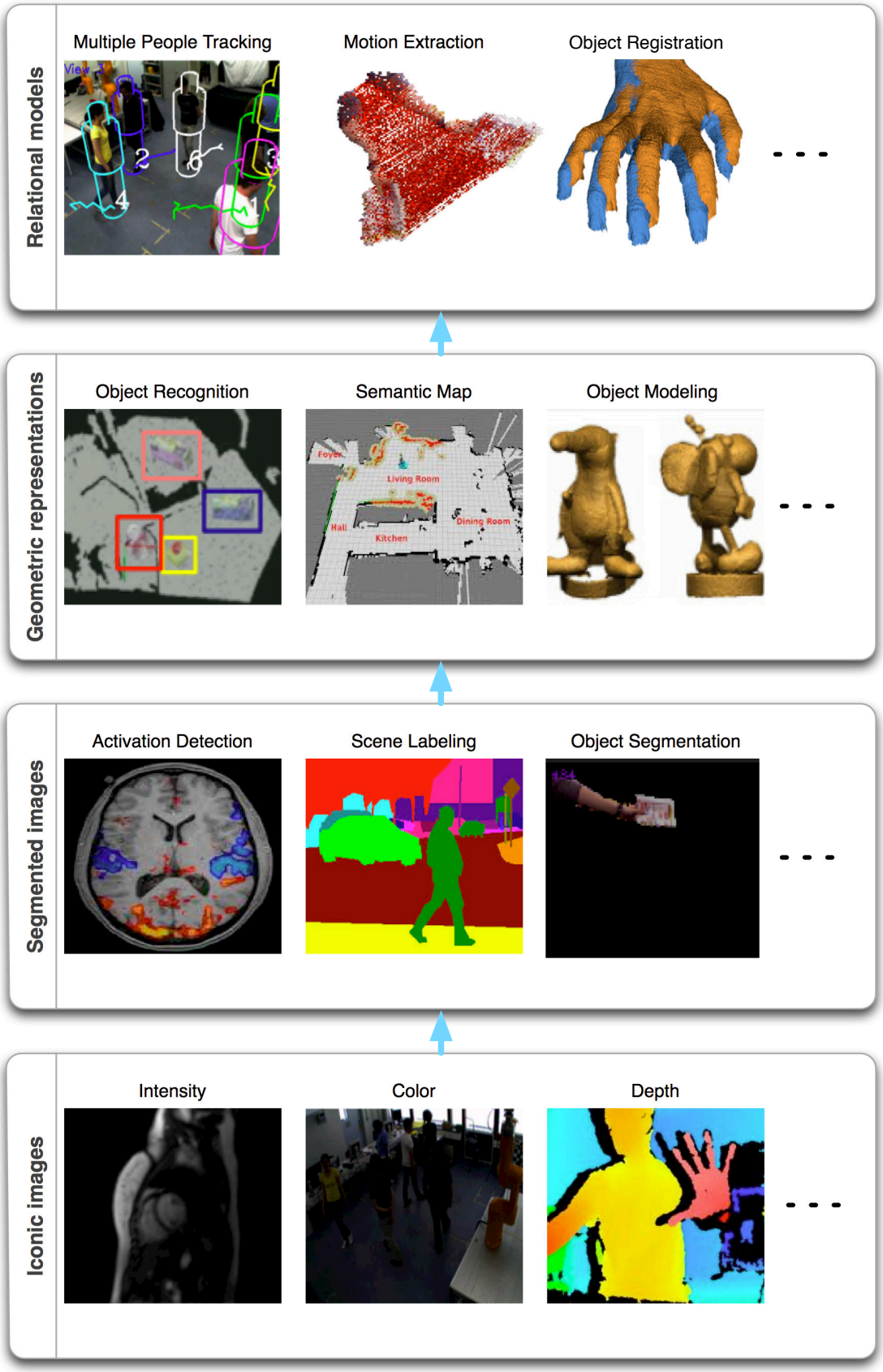


Figure 1.2: Different levels of visual data structures: iconic images, segmented images, geometric representations and relational models. Typical application examples are illustrated for each visual data structure.

depth. Useful information can be extracted by highlighting some aspects of the image important for further treatment using operations such as filtering and edge sharpening. At the next higher level visual data structures as "segmented images", parts of the image is separated into regions based upon their classification. This can be used to obtain a compact scene representation, extracting only the "interesting" areas and reducing complexity. In many applications, objects need to be represented using a comprehensive and efficient descriptor. The latter can be based on "geometric representations" of the object that can be extracted from the segmented image. The last category refers to "relational models" that provide the means to treat data at a very high level of abstraction. These models are obtained leveraging a priori knowledge about the application as well as a number of assumptions and constraints, e.g. environment context, physics models, motion constraints and group relationships. This kind of prior knowledge is widely applied in different computer vision applications, for instance, multiple people tracking, dynamic motion extraction, surface deformation analysis etc. The relationships of these four level visual data structures and some vision applications are illustrated in Figure 1.2.

Finally, the last component of the perception system, "understanding" provides users with reasoning information utilizing matching and optimization processes. Based on the analysis of properties of visual data structure, it provides the capabilities for understanding and recognition of dynamic scenes. For instance, the perception system of an autonomous robot can adapt its actuators from the understanding results, which are geared towards multiple robotic applications.

When planning a perception system for a specific vision application, the sensor subsystem is first designed to provide all the raw information necessary to represent the surrounding dynamic environment. Based on processing and analysis of the raw data, visual data structures at different levels of abstraction are then constructed. The properties of the visual data structures are then used to extract the relevant high level information that is the output of the perception system. For example, in the problem of tracking multiple people from a RGB camera, the raw data are the pixels of each video frame. The moving pixels can then be segmented from the background scene based on a color image. The structure properties are then used to define an object descriptor based on the geometric relationships between individual features. Each such descriptor corresponds to a person that can be detected, localized and tracked. At a higher level, motion constraints and other assumptions will be used to make the tracking more robust and to allow group tracking and behavior analysis.

1.2 CHALLENGES AND OBJECTIVES

To build a successful perception system several challenges still exist and significant improvements need to be implemented within the different components:

- How can a suitable sensor system be built that can collect comprehensive information representing the surrounding dynamic scene, and matching the application requirements?
- How can useful information be efficiently extracted from visual data structures at different levels of abstraction?
- How can this lead to an improved understanding of the dynamic environment?

From the technical point of view, a perception system for scene understanding and recognition has to find a good trade-off between robustness, accuracy, and efficiency. When such a system is used in different vision applications it has to reach different goals accurately and robustly despite sensor noise and ambiguous observations.

In this thesis, we focus on the research area of scene understanding and recognition from visual data structure properties in dynamic environments. This research is divided into four different vision application areas. Their objectives are separately described as follows:

(i) *Real-time object recognition and pose estimation for robot's manipulation and exploration*
The objective is to recognize and determine the 6D pose of textured/textureless objects in 3D dynamic environments in real time;

(ii) *3D Motion estimation and analysis for dynamic scenes*
The objective is necessary to extract the dense and deformable motion in 3D space and retrieve the detailed surface deformation information simultaneously;

(iii) *Articulated object recognition and manipulation for autonomous robots*
The objective is to recognize deformable object can be recognized within different configurations in dynamic environment. And at same time, the robot can adapt proper manipulation skills based on relevant joint state and task goal ;

(iv) *Real-time human body motion estimation for outdoor autonomous exploration robots*
A suitable sensor system needs to be built for outdoor robotic vision applications. The objective is to estimate the accurate human body part position, motion and full human pose in real time.

1.3 MAIN CONTRIBUTIONS

In this thesis we aim to tackle the problem of scene understanding and recognition from visual data structure properties in dynamic environments for several application areas. In each of these areas we bring a number of contributions as detailed below:

3D Object Recognition and Pose Estimation: we propose a real-time object recognition and pose estimation system for autonomous robot exploration and object manipulation. This system can recognize both textured and textureless objects, as well as accurately determine their 6D pose in a 3D dynamic environment. The main contributions of this research include:

- A novel object descriptor called *Viewpoint oriented Color-Shape Histogram* combining color and shape features, as well as information about the camera viewpoint;
- A real-time object recognition and pose estimation system which gives high recognition rate and accurate 6D pose recovery under various unstructured environments;
- 3D object recognition and localization for coherent semantic mapping;
- Performance evaluation and state-of-the-art comparisons on object recognition rate, pose accuracy and stability analysis with respect to illumination changes;

Dense and Deformable Motion Estimation: we propose a motion estimation and analysis system for dynamic scenes that can extract dense and deformable motion in 3D space. Point-level motion is estimated and detailed surface deformation information is retrieved from the spatial and temporal properties of the visual data structures. The main contributions in this area include:

- A novel hierarchical MRF structure for 3D dense and deformable motion extraction. It consists of segmentation and correspondence layers, which are formalized as an image pixel-level and a 3D point-level MRF;
- Novel global energy functions for optimization of the segmentation and correspondence layers in our hierarchical MRFs;
- A new deformable surface descriptor *Deformable Color and Shape Histogram* combining photometric and geometric information;
- A dataset of dynamic RGB-D scene sequences featuring different motion patterns and surface properties of the dynamic foreground;
- Performance evaluation on segmentation and correspondence accuracy, runtime performance and comparison of different optimization schemes.

Articulated Object Recognition and Manipulation: we design an articulated object recognition and manipulation system for autonomous robots. The deformable object can be recognized regardless of its current configuration. At the same time, the robot can apply appropriate manipulation techniques based on the task and the state of the articulated joints of the object. The main contributions include:

- A framework for single-joint articulated object recognition, joint state estimation and robotic manipulation;
- A novel articulated object modeling method combined with visual and manipulation observations;
- A depth-based skeletonization method to extract visual observations of articulated object;
- Determine number and type of joints as well as their working space constraints;
- Learning of manipulation tasks by demonstration that is mapped into the articulated object joint space.

Real-Time Human Motion Estimation: we propose a real-time human body motion estimation system for outdoor autonomous exploration robots. A suitable sensor system is built for outdoor robotic vision applications. With the extraction of useful information representing human motion, this perception system can accurately capture the position and motion of human body parts as well as the full human pose in real time. The main contributions in this area include:

- A framework for real-time human body motion estimation based on multi-layer laser scans;
- A novel method of *Iterative Template Matching for Clustering and Segmentation* (ITMC) to extract the human visual features;
- Human contour extraction from multi-layer 2D laser sans;
- Mapping of extracted features with pre-specified articulated human skeleton model in real time;
- Motion accuracy and runtime performance evaluation and comparisons;

In all of the aforementioned areas, the perception system is designed to capture raw visual data (RGB, depth images and 2D points) into a hierarchical set of data structures. The properties of these visual data structures are then analyzed and used for extracting application-specific high-level information. Although in this thesis, we propose the application in four areas, this processing enables a further number of different applications in both academic as well as industrial domains.

1.4 OUTLINE OF THE THESIS

The remainder of the thesis is organized as follows. Chapter 2 provides an overview of the related work in the field of object recognition and pose estimation, dynamic scene motion extraction, articulated object recognition and manipulation, real-time human motion estimation and other research that influenced this thesis.

Chapter 3 presents a real-time textured/textureless object recognition and 6D pose estimation system, and extension for object localization in a coherent semantic map. The global object descriptor named Viewpoint oriented Color-Shape Histogram is described here to represent rigid object patch data that is oriented by camera viewpoint. We also present the strategy to build the object model and generate object patch data from different synthetic viewpoints. An object recognition scheme and a pose optimization method are illustrated afterwards. In the following we present a large number of experiments, including daily complex scenarios and indoor semantic mapping. The detailed evaluations and state-of-art comparisons are presented in the end.

Chapter 4 presents a novel hierarchical MRFs optimization method for dense and deformable motion extraction from dynamic RGB-D scenes. In particular, we show the details of this hierarchical MRFs structure consists of two layers, respectively the segmentation and correspondence layer. A new surface descriptor, named Deformable Color and Shape Histogram, is proposed. The discrete optimization scheme is utilized for these binary classification and multi-labeling problems. Moreover, a dataset containing common dynamic RGB-D scenes is introduced in general, which involves different motion patterns and surface properties of dynamic foreground. The evaluation of accuracy and runtime performance are illustrated to validate the proposed method.

Chapter 5 presents an approach to model articulated objects by integrating visual and manipulation information. Firstly, we illustrate a new method of line-shaped skeletonization based on depth image data which extracts the skeleton of an articulated object in different configurations. A method for the characterization and localization of the joint types is then presented. Followed by a description of how to learn to learn a robotic end effector's force data in terms of the task-space force required to manipulate the object, into estimated object kinematic joint state space. In the end, the experimental results of multiple demonstrations are described and the effectivity and efficiency of our propose articulated object modeling method is validated.

Chapter 6 presents a method for real-time 3D human body motion estimation based on three-layer laser scans. All the useful scanned points, presenting the human body contour information, are subtracted from the learned background of the environment. A novel iterative template matching algorithm for segmentation and clustering is proposed. We also present a method for the robust extraction of distinct motion features

using a maximum likelihood estimation and nearest neighbor clustering. Subsequently, the method is illustrated in detail, which associates the extracted features with a predefined articulated model of a human body to retrieve the positions of the human joints in 3D space. Finally, the experimental results are presented with evaluations and comparisons.

Chapter 7 concludes this thesis, listing the key contributions and detailing a number of interesting issues that are left for future work. Some extensions of our approaches respect to other related computer vision and robotics issues are also discussed.

2

RELATED WORK

In this chapter we provide the context behind this research. An exhaustive literature review in the topic of scene understanding and recognition is impossible due to the large volume of published work in the area. As a consequence, this chapter will focus on work that has had a significant impact on this field or is closely related to the work presented in this thesis. We present an overview of the most recent advances in the areas of object recognition and pose estimation, motion extraction of dynamic scene, articulated object recognition and manipulation, and real-time human motion estimation. Strengths and potential deficiencies are discussed through a comparative analysis, which enables us to identify the key points that need to be considered by this research.

2.1 3D OBJECT RECOGNITION AND POSE ESTIMATION

To interact with autonomous robots in unstructured environments, it is essential for a robot to successfully recognize objects, estimate its accurate pose and perform high-level tasks in real time (Figure 2.1). Therefore, object recognition and pose estimation plays a crucial role in a wide range of robotics applications. It is also at the heart of other high-level tasks such as object localization for semantic mapping. However, it presents a very challenging problem due to the large variability in respect to object size, position and viewpoints, as well as the heavily cluttered environments and/or the occlusions in the scene (see Figure 2.2) [15, 16, 37, 38, 41].

Some previous approaches have been developed to address the challenges mentioned above. Among these, an efficient object descriptor plays a very critical role. There is a large variety of object descriptors using diversified features. For 2D images, SIFT [92], SURF [12], HOG [31] and BRIEF [21] are the most popular descriptors that can be extracted based on the photometric properties (texture) of objects. In addition to gray-scale features, color-based features have also been widely proposed for object recognition [1, 30, 47, 48]. However, the photometric features have the limitation of not being able to cover all potential poses in 3D space. In the case of 3D depth images, a wide variety of geometric quantities have been used to emulate comparable features, in order to be used for geometric descriptors. These include local patches [96], local moments [29], volumes [46], polygon surfaces [121], spherical harmonics [129], con-

tours [106] and edges [81]. However, these geometric features only describe 3D object shape primitives while ignoring photometric information.

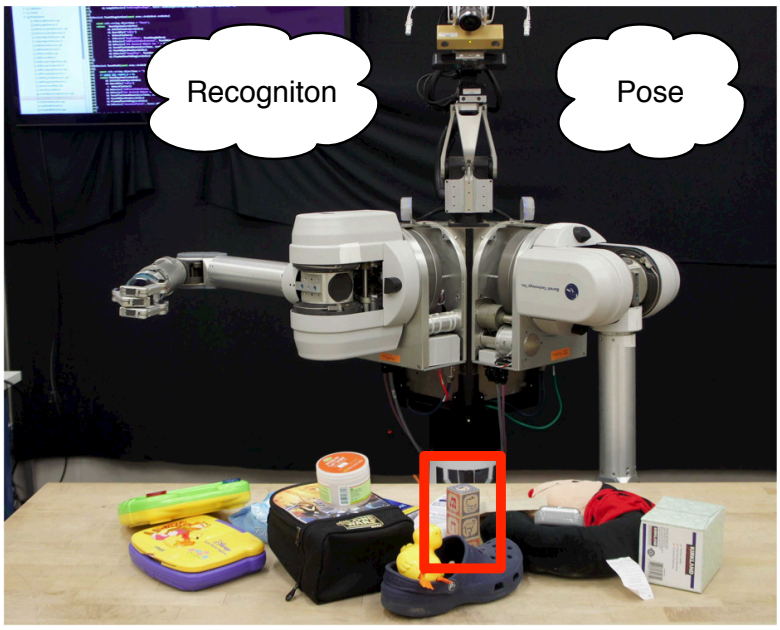


Figure 2.1: An autonomous robot needs to recognize objects in its view and get their 6D pose in unstructured environments.

Lately new RGB-D sensors such as the Kinect and stereo cameras have been introduced. These sensors can provide both photometric and geometric information at the same time. Object descriptors based on multi-dimensional photometric and geometric features provided by the aforementioned sensors are a powerful alternative for object recognition and pose estimation. Such descriptors have been considered for example in [27, 73, 74, 146, 151]. Furthermore, an object should be recognized regardless of its pose (scale and rotation invariant). To achieve this the viewpoint component needs to be integrated into the object descriptor [39, 62, 82, 87, 119, 148].

A large variety of approaches have been proposed for object recognition and pose estimation. Within those approaches, object descriptors are mainly classified into two categories: global and local. Global object descriptors extract features from well segmented and clustered object data [119, 146, 167]. The object needs to be well clustered and it is sensitive to partial occlusions. On the other hand, a local descriptor is based on a pair-to-pair feature matching strategy from real-scene data which results in a high computational cost for final recognition and pose recovery [27, 74, 118, 150, 151].

More specifically for global object descriptors, new VFH [119] as an extension of FPFH [118], integrates the viewpoint variant component into the 3D geometric features. However, it neither allows for full pose estimation nor considers texture or color features. Wohlkinger et al. propose a global 3D descriptor named Ensemble of Shape Functions (ESF) [167]. ESF creates a dataset by generating synthetic views using CAD

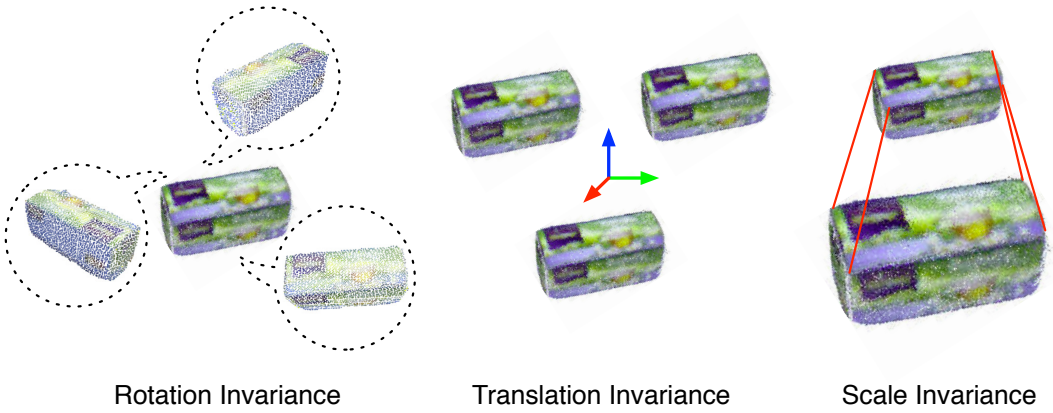


Figure 2.2: The object descriptor needs to be rotation, translation and scale invariant.

object models. The combination of angle, point distance and area shape are applied on randomly selected point pairs, while local distribution features are accumulated into a global descriptor. Nevertheless, ESF neglects the object's photometric information and is thus unable to accurately provide pose estimation for certain types of objects. Tang et al. [146] directly use the Naive Bayes matching method for object recognition and pose recovery. The object global hue value histogram is generated from the complete mesh object model. Combined with the extracted 3D SIFT from object's texture, the object can be recognized and its pose can be estimated. However, this approach needs a detailed object mesh model for training. Moreover, objects need to be fully textured.

For local object descriptors, Signature of Histograms of Orientations (SHOT) divides the spherical volume around one point into spherical grids based on the local reference frame [151]. The normal of each point falling into a certain grid is compared with the normal of the centroid. The angle relationship is measured and represented as a histogram on each grid which is then concatenated as a descriptor. CSHOT is an extension of SHOT that adds color information during the construction of the descriptor and is presented in [151]. This method relies on a local reference frame, but the reference frame cannot be robustly estimated for objects with rotational symmetry (e.g. a basketball). A real-time object recognition system is proposed in [74]. It uses ConVOSCH object descriptors which correlate geometric and visual RGB data, but is unable to accurately obtain the object's pose. Choi et al. [27] define a local object color point pair feature descriptor, which is represented as a hash table combining geometric and HSV color information. However, the color information is only utilized for pruning potential false matches and is not considered as a general object descriptor for recognition and pose estimation. Moreover, this approach results in a high computational cost. And the result quality strongly relies on high dimensional parameter settings, which needs to be adjusted respect to different scenes.

2.2 DENSE AND DEFORMABLE MOTION EXTRACTION

Dense and deformable motion extraction in dynamic scenes is arguably one of the most interesting and important fundamental problems in many computer vision applications. The 3D motion field in the dynamic time-varying image sequence can be represented as a dense displacement vector field that links the locations of each image point across consecutive image frames. Motion estimation provides a comprehensive understanding of the dynamic scene. It is also crucial for a number of computer vision tasks such as object segmentation [49, 60, 65, 112], object tracking [18, 25, 75, 134, 170], human motion estimation [32, 56, 135, 175], etc., as shown in Figure 2.3.

However, after decades of research and development on motion extraction, there are still several challenging situations that remain unaddressed. These includes cases where: 1) the dynamic scene contains different motion patterns; 2) the dynamic foreground data has different surface types like rigid or deformable; 3) different features need to be extracted and correlated effectively and efficiently; 4) a difficult balance needs to be struck between accuracy and computational cost.

The literature provides several methods directly related to dynamic scene motion extraction. These methods are mainly classified in two categories: transformation model based and correspondence matching based.

In methods based on transformation models, each point is considered to be constrained in its possible displacement across consecutive frames [53, 61, 67, 111, 120, 158]. Optical flows [128], is perhaps the most popular one, and was originally proposed for the 2D motion field estimation based on image changes. To deal with the limitation of large motion estimation of optical flows, Brox and Malik [18] incorporated local descriptor matching to improve differential optical flows. Wedel et al. [165] extend the work of [18]. They proposed decoupling of the motion estimation from the disparity estimation while maintaining the stereo constraints to calculate the 3D motion in real time. With the combination of depth and color information, scene flows are presented as an extension of optical flows to represent the 3D motion of points [60, 156, 157, 175]. However, methods based on transformation models have the drawback that they cannot capture the true deformation of surface details. This method is therefore not suitable for motion extraction of a deformable foreground with large displacements. Such a situation appears often in dynamic scenes, for example when the scene contains a moving soft cloth surface.

Opposite to the transformation model based methods, correspondence matching methods extract dynamic motion based on global point-point matching [77, 159]. Without a transformation assumption, a correspondence map is extracted only based on similarity matching and regularization by different constraints. The dynamic scene motion can be retrieved, once the displacement of a point with its correspondence across

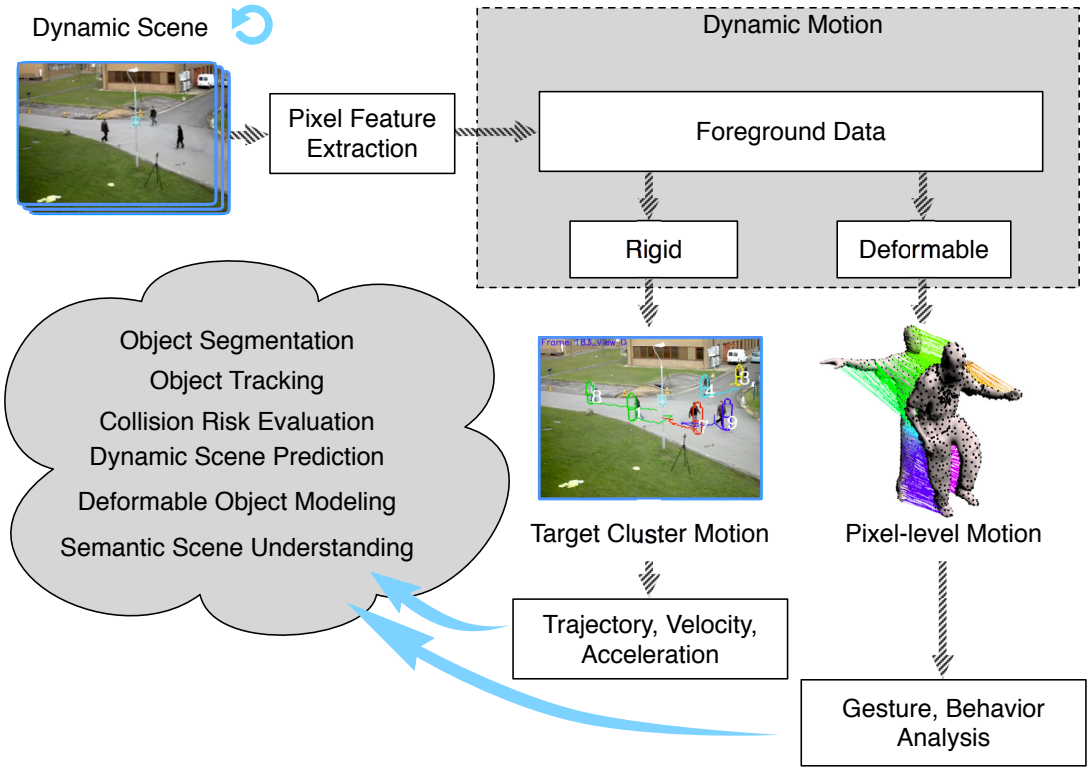


Figure 2.3: Different type of dynamic motion as rigid dense motion or deformable motion. The extracted motion can be used for different applications.

consecutive frames is established. In [4], SIFT is used to obtain sparse correspondences between adjacent frames under the assumption of isometric deformations. Zaharescu et al. [171] extend 2D local features to 3D feature detectors and descriptors to improve sparse matching of meshes. The descriptors capture photometric properties as well as local geometric properties. However these keypoint-based matching cannot deal with a homogenous foreground. Instead, all surface points correspondences are considered to model the detailed deformable surface motion. Dense point clouds are registered using a randomized feature matching algorithm relying on geodesics [147]. Tung et al. [153] present an approach for dense matching of dynamic surfaces in 3D videos using geodesic maps. But this technique does not involve any photometric information, while at the same time requiring the existence of prior models and resulting in a heavy computational cost. Thus, we consider a hierarchical strategy to extract an interesting area at first followed by a matching of points in this extracted area instead of using entire image. This strategy can extract the detailed surface motion with a lower computational cost. The work most similar to ours is by Zhang et al. [174]. They proposed the extraction of human motion by tracking all 3D points of a deformable object without any transformation assumptions. They emphasize that their work is the first to

track all point clouds for motion extraction. However, photometric information is not considered in this work. Their coarse-to-fine procedure relies on the assumption that the matching candidates are searched only in a certain area. This method also does not prevent multiple points to be matched with a single point. Moreover, it does not focus on dynamic motions and the final correspondences are not sufficiently accurate to calculate detailed deformations.

In general, most of the related work in this area is application-dependent, relying on prior knowledge or strong constraints to simplify problem. For instance, rigidity, small displacement and/or elastic deformation constraints may be imposed [125, 172]. Other approaches have assumed pre-defined accurate object models [98, 137], full texture, and so on.

2.3 ARTICULATED OBJECT RECOGNITION AND MANIPULATION

Most daily tasks require manipulation of articulated objects of one or more degrees of freedom. Some characteristic examples of such tasks are opening doors, opening/-closing drawers or rotating a water tap. Manipulation of articulated objects is a great challenge for autonomous robots. They are required to recognize an articulated object often using vision and make a decision about how to manipulate it. Robots that have the aforementioned capabilities can be used for helping humans in daily tasks or for taking over dangerous/difficult tasks without needing to modify the objects to be manipulated. As shown in Figure 2.4, the Murola robot[1] is trying to manipulate a car door (open and close). This door can be viewed as an articulated object with single rotational joint.

An articulated object is defined as a deformable object composed of a kinematic chain connecting different rigid parts. Figure 2.5 shows examples of articulated objects. Previous research on articulated object modeling mainly focused on solving the problem of identifying the kinematic characteristics of articulated objects using different types of sensor systems [17, 138, 140, 141, 166].

To recognize an articulated object from visual information, it is necessary to extract features or properties for different object configurations. For example, texture features or marker-based detection processes have been used to get the deformation trace of articulated objects [17, 138, 141]. These methods are limited to certain object visual information types (textured) and are inefficient when solving the rotation invariant problem. Instead, Pellegrini et al. use a generalization of interactive close point (ICP) to estimate the articulated structure. On the other hand, Sturm et al. use a depth image to classify the articulation type and predict the motion [139]. Nevertheless, these

1 http://www.lsr.ei.tum.de/en/research/areas/robotics/murola-the-multi-robot-lab/

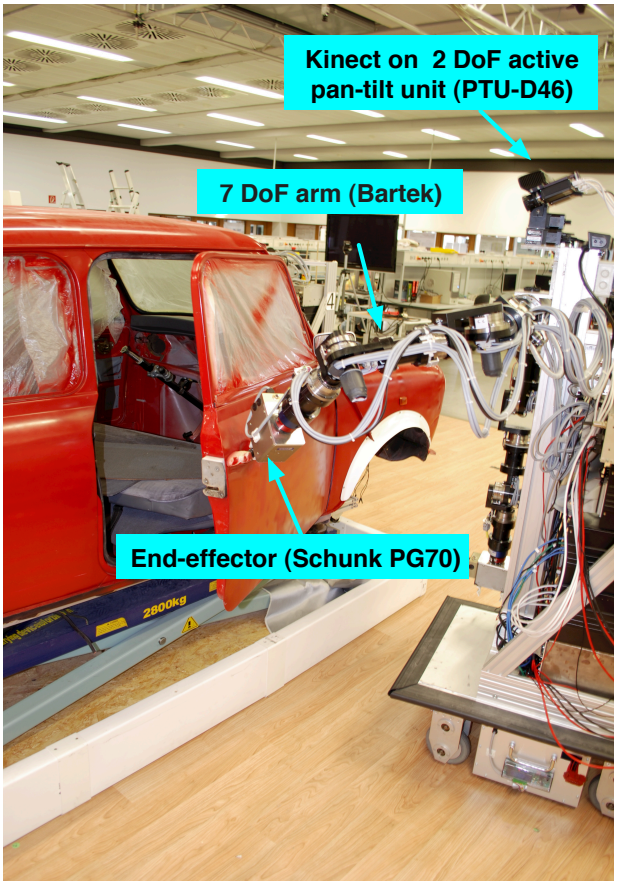


Figure 2.4: A 7 DoF robotic arm manipulates a car door.

approaches are based on either prior knowledge on object model or rigid surface assumptions. Weiss et al. present a method for representing and recognizing objects based on range images [166]. Their work uses the object's regions to estimate invariants to deal with low resolution images, when it is hard to use common features such as edges. Sturm et al. use visual markers to estimate the configuration of articulated objects [138], In their follow-up work [140], an approach is presented to learn kinematic models of articulated objects from observations. They also apply same learning approach for articulation classification and motion prediction of objects based on depth images [139]. The object is under an assumption of planar surface. This method can estimate the pose and motion of an single-joint articulated object. Research in articulated object tracking has also focused on tracking the human body [34, 45, 154, 181]. These methods handle a specific kind of input (e.g. video only) or attempt to learn motion models to improve performance. Brookshire et al. use particle filters for articulation pose estimation via over-parametrization and noise projection [17]. Their proposed system works under the assumption that patch maker positions for the object have been given. In general, the previously reported approaches do not allow for object identification and ignore kinematic joint localization as well as its working space constraints.

Figure 2.5: Some examples of articulated object.

An object skeleton has been widely used for many vision applications, such as human pose estimation [26, 28, 59, 71, 78, 124, 136] and object recognition [141, 166]. Object skeletonization is mostly based on the geometric analysis of complete 3D object models [58, 91, 110, 127, 149, 168, 178], center line extraction from contour information [10, 114, 164] and predefined skeleton models [26, 59, 124, 136].

Katz et al. [76] use kinematic task-relevant knowledge and generate manipulation skills in the joint state space of the articulated object. This is realized via interaction with the environment and, finally, a kinematic model of the object is incrementally built. However, only visual data is employed and information about the dynamic properties of the object is not taken into account for manipulation. In [66], the position of the joint axes of an articulated object is estimated given different object configurations from depth image data. This aims at providing grasping points and a position trajectory to the robot. However, recognition of the object configuration is not considered. All of these approaches lack a framework for recognition of different articulated objects. In addition, they do not allow the estimation of the current joint state in order to adapt the manipulating behavior accordingly. In addition, previous research did not account for learning the force that is required to operate an object. For example, opening a completely closed and opening a semi-closed door are two different tasks which require different manipulating forces. On the other hand, other researchers have focused on manipulating articulated mechanisms by learning force control skills while ignoring the mechanism structure of the object. In all these works [72, 93], no visual information is used to recognize the object, the number and type of joints or the constraints that apply on each joint of the object. Therefore, these approaches cannot generalize to objects with different structures or configurations.

To the best of our knowledge, most of research works on articulated objects focuses either on using visual data for object characterization without learning manipulation force [70], or on learning manipulation force skills without analyzing the articulation characteristics of the object. Learning manipulation of even a single-joint articulated object is a challenging problem, since the articulation characteristics of the object have to be extracted first before an appropriate manipulation force is learned.

2.4 REAL-TIME HUMAN MOTION ESTIMATION

Nowadays human body motion capturing [8, 11, 132, 180] and pose estimation [2, 40, 54, 64, 84, 100, 103, 126, 169, 177] have attracted increasing attention due to their widespread use in human robot interaction, the analysis of human social behavior and other applications for service robots [105, 116]. Nevertheless, several challenges still exist: 1) Because of sensor technology limitations, it is difficult to choose the proper sensor to acquire enough visual information for human; 2) The information from sensors is limited for extracting a sufficient number of features; 3) Visual feature extraction based on raw information is hard to be efficiently obtained to represent the cues of human pose and motion; 4) It is difficult to correlate and associate these visual features with human model, to estimate human motion in real time.

In the early days of human pose and motion estimation research, 2D images were first considered. Fujiyoshi et al. [44] use skeletonization technology from human image boundaries to get human motion in a video stream. Aiming to get 3D human motion, Sidenbladh et al. [131] use a probabilistic method for tracking 3D articulated human motion in monocular image sequences. Hofmann et al. [63] use video surveillance from four corner cameras to estimate the 3D human pose. Iwashita et al. [68] propose a model-based motion tracking system using distributed network cameras that are placed in a sizeable environment. These methods limit applications to a certain working space or cannot retrieve accurate 3D human motion. Hence, these methods are not well suited for the real-time human motion estimation system of an autonomous robot.

Some researches have used depth cameras to estimate human motion. Time of Flight (TOF) camera based approaches are utilized for online 3D human body motion capturing, due to the richness of the information obtained [79, 108]. With the development of new sensor technologies, some sensors find increased use in robot perception system. Such a sensor is the Microsoft Kinect[2], which can provide full 3D view depth and color information. Ye et al. [169] present real-time human body pose estimation from a single Kinect. This was achieved by matching the frame image configuration with pre-captured motion exemplars. To improve the tracking quality and reduce ambiguities as for example caused by occlusions, Zhang et al. [173] fuse the depth images of all available cameras into one joint point cloud to track the high-dimensional human pose.

Nevertheless, an important issue with these kinds of sensors is that they are quite sensitive to illumination and other environmental changes, which is why most of them are still limited to indoor applications. Compared to these sensors, laser range finders (LRF) are much more robust to illumination changes [22, 99, 109]. They also provide a

2 http://www.primesense.com

Figure 2.6: Autonomous city explorer robot ACE.

large scan range, a high data rate, and accurate measurement that makes them especially suitable for outdoor applications such as autonomous city explorer robot ACE[3] (see Figure 2.6). However, the disadvantage of standard LRFs is that the obtained information is 2D, which is limited compared to the aforementioned sensors.

There are two ways to overcome the 2D limitation of LRFs: actuation or use of multi-layer scanning. As demonstrated in [94, 104], it is possible to achieve good 3D scans of static environments by actuating LRFs. However, it is usually quite time-consuming to make a full scan which makes this approach unsuitable for real-time tasks such as human tracking. Multi-layer laser scanning does not have this problem and can be obtained either using multiple single-layer LRFs or a sensor with built-in multi-layer scanning, such as the ibeo-LUX[4].

Multi-layer LRFs system has been widely utilized for accurate people detection and tracking [22, 109]. Mozos et al. [109] use a static 3-layer LRFs system to detect the surrounding people. This approach is composed of a probabilistic combination of the outputs from different classifiers which are extracted from each layer LRF. These independent classifiers provide the detections of particular body parts including head, torso and leg. Carballo et al. [22] mount a 2-layer LRFs system on autonomous robot to estimate human position. In this work, a predefined human model is built to estimate the specifical person's position in scene, by associating multiple features, e.g. area of

3 http://www.ace-robot.de/
4 http://www.ibeo-as.com/

chest and leg and a volume representation. Nevertheless, these works only focus on people position estimation and tracking. Moreover, there exist some research works to estimate people position as well we pose using multi LRFs [50, 95]. Glas et al. [50] propose a people tracking method using particle filter to get not only the people location but also the body rotation and arm position, by using the contour information from the torso-level lasers. Matsumoto et al. [95] try to use multi LRFs to get the contour features for multiple people pose estimation. These LRFs are located at same height and in four corners of a certain area. Some predefined pose candidates are weighted get the best matched pose as the final result, based on a resampling process and propagation by a transition model. The estimation region is limited and only certain number of poses can be estimated.

The aforementioned related works based on multi-LRF setups are only used for either people detection and tracking, or estimation of certain number of human poses. Laser range finder has its benefits like fast, wide range, robust and wide working space. However there still exists a drawback, that only a fewer amount of information is obtained compared with other full 3D view sensors. Consequently, there exist huge challenges to estimate real-time human body motion based on the multi-layer LRFs mounted on an autonomous robot.

2.5 SUMMARY

In this chapter, a literature review in the fields of scene understanding and visual recognition for dynamic environment has been provided. The detailed background and related works which have had a significant impact on these fields or are closely related to our works are presented for following sub-topics: 3D object recognition and pose estimation; dense and deformable motion estimation; articulated object recognition and manipulation; real-time human body motion estimation. We have discussed the advantages and potential deficients of the related works and potential deficiencies according to the problems that we aim to solve.

In the following chapters, we will present the details of the proposed methods, system structures, experimental results and conclusions for these four subtopics.

3

REAL-TIME 3D OBJECT RECOGNITION AND POSE ESTIMATION

In this chapter, we propose a novel global object descriptor, called *Viewpoint oriented Color-Shape Histogram* (VCSH), which combines 3D object's color and shape features. The descriptor is efficiently used in a real-time textured/textureless object recognition and 6D pose estimation system, while also applied for object localization in a coherent semantic map. We build the object model firstly by registering multi-view color point clouds, and generating partial-view object color point clouds from different synthetic viewpoints. Thereafter, the extracted color and shape features are correlated as a VCSH to represent the corresponding object patch data. For object recognition, the object is identified and its initial pose is estimated through matching within our offline generated dataset. Afterwards the object pose can be further optimized utilizing an iterative closest point strategy. Therefore, all the objects in the observed area are finally recognized and their corresponding accurate poses are retrieved. We validate our approach through a large number of experiments, including realistic, complex scenarios and indoor semantic mapping. Our method is proven to be efficient and guarantees a high object recognition rate, accurate pose estimation, It is also capable of dealing with environmental illumination changes.

The remainder of this chapter is organized as follows: Section 3.1 to Section 3.4 provide a detailed description of the VCSH descriptor, its integration within the object recognition and pose estimation system as well as for object localization in semantic maps. The experimental results including the pose accuracy evaluation, stability analysis under illumination changes and runtime performance experiments are presented in Section 3.5. Finally, Section 3.6 summarizes this chapter.

3.1 FRAMEWORK FOR OBJECT RECOGNITION AND POSE ESTIMATION

In this section, we provide details on the design of our VCSH descriptor, and how it is integrated into an object recognition and pose estimation system, and finally show that it is efficiently applied for object localization in semantic mapping.

The framework of our proposed approach is illustrated in Figure 3.1. During the offline training phase, we first build the complete 3D object model by registering all of the object's RGB-D data in different poses into a single coordinate frame. By using the centroid of the object model as the origin, we generate a sphere with a certain radius. On the surface of this sphere, a large amount of viewpoints are homogeneously

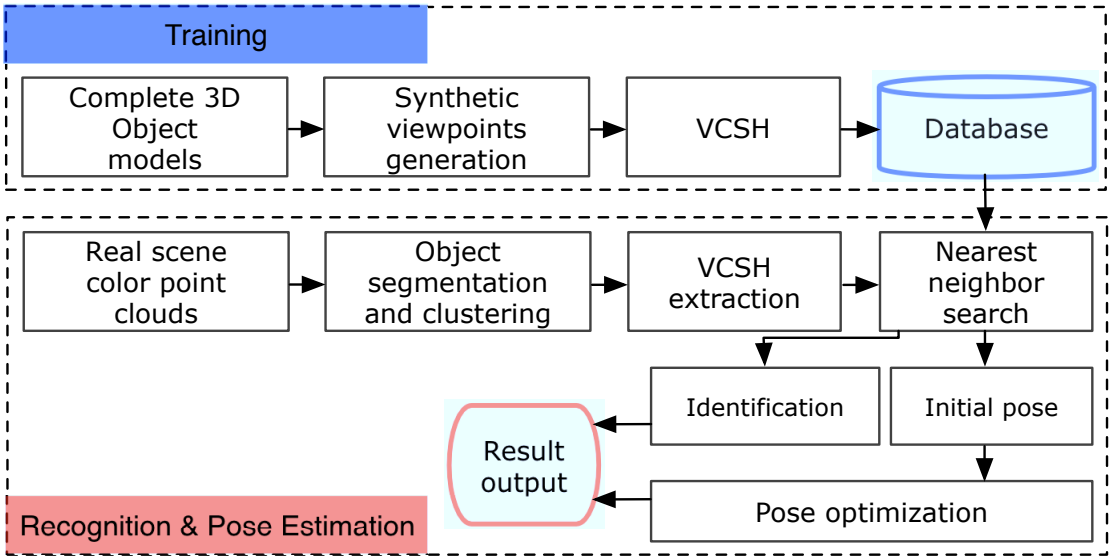


Figure 3.1: Overview of a real-time textured/textureless object recognition and pose estimation system using the viewpoint oriented color-shape histogram (VCSH) descriptor.

generated with their direction pointing to the sphere origin. Using each of these generated viewpoints, object patch data representing the object identification and its corresponding viewpoint pose, are generated. Subsequently VCSH can be computed as a global object descriptor for all object patch data, within which the color and shape information of all points is used for the descriptor generation. Consequently, an object is represented by the generated VCSH set and stored into the dataset. During the online recognition and pose estimation phase, the object data is segmented and clustered from the real world scene, and we compute its corresponding VCSH. Thereafter, the likeliest candidate is retrieved from our generated descriptor dataset by nearest neighbor searching. Using the initial candidate pose, the recognized object's accurate 6D pose can be estimated through a pose optimization and verification step. In addition, the object recognition and pose estimation system is applied into the coherent semantic map, for the robotic exploration in large-scale map and for further object manipulation. Next we explain in greater detail the parts involved.

3.2 3D OBJECT MODELING AND VIEWPOINT ORIENTED PATCH GENERATION

3.2.1 *3D Object Modeling based on RGB-D images*

Our proposed object model building platform consists of a rotatable plane and a stationary Kinect sensor. After segmentation from the plane and Euclidean distance-based clustering, object color point cloud data $\{O_f\}$ for each single view and its transform $\{TF_f\}$ relative to the initial frame O_0 are captured, where $f = \{0 \cdots F\}$ is the frame

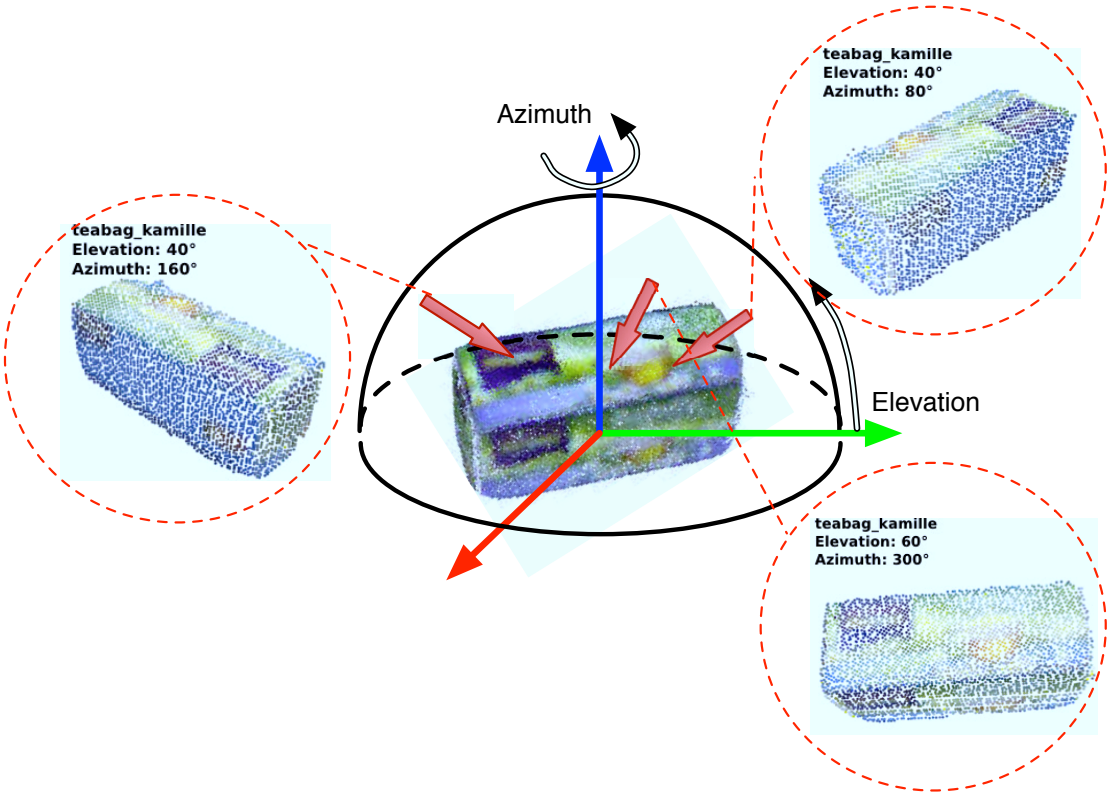


Figure 3.2: Sampling the synthetic viewpoints in the upper hemisphere for object patch data generation: Red vertices represent the virtual camera viewpoints and the red circles illustrate some generated data from synthetic viewpoints.

index. By registering $\{O_f\}$ with $\{TF_f\}$ into a single object coordinate, the whole 3D model Ω can then be generated as a cluster of color point clouds,

$$\Omega = O_0 \cup TF_1^{-1} \cdot O_1 \cup \cdots \cup TF_F^{-1} \cdot O_F. \tag{3.1}$$

In order to eliminate noise, the Moving Least Squares (MLS) algorithm [6] is utilized to smooth the entire 3D model. Note that the detailed object mesh model and surface texture information are not necessary here.

3.2.2 *Different-view Object Patch Extraction from Synthetic Viewpoints*

For each object model Ω_i, $i = \{1...I\}$, we generate J object patch data M_j with synthetic viewpoint VP_j where $j = \{1 \cdots J\}$. Note that the viewpoint is the sensor's view direction relative to the object. Since the view direction needs to cover all potential 6D poses of the object, the synthetic viewpoints are therefore generated on a half sphere surface, with the origin being the centroid of the object model. The synthetic viewpoint position is generated on the sphere surface homogeneously both in the elevation and azimuth directions, with its direction pointing to the sphere's origin. With the gener-

ated synthetic viewpoint VP_j, object patch data M_j can be generated according to VP_j from the whole 3D object model Ω by using the ray-casting method, as illustrated in Figure 3.2. A pseudocode implementation is also given in Algorithm 3.1.

It is necessary to mention that the object model Ω is not only restricted to the raw color point cloud model, but can also be obtained from CAD models.

Subsequently, a global object descriptor is needed to describe each M_j with its viewpoint VP_j for object recognition and 6D pose recovery.

Algorithm 3.1 Object patch data generation using sampled synthetic viewpoint

```
1:  Model Ω;                                  # whole 3D object model
2:  Data M;                                   # generated object patch data
3:  Viewpoint VP;                             # related synthetic viewpoint
4:  Double ε;                                 # threshold for point in a line
5:  for i := 0 to Ω.pointsize step 1 do
6:    p ← Ω.points[i];                        # point in Ω
7:    L ← line3D(VP, p);                      # get the relative 3D line
8:    Flag ← false;                           # flag of occluded
9:    for j := 0 to Ω.pointsize step 1 do
10:     if i ≠ j then
11:       p* ← Ω.points[j];
                                              # another point in Ω
12:       if dist(p*, L) < ε and ||VP − p*|| < ||VP − p|| then
13:         Flag ← true;                      # point in line and closer to viewpoint (occluded)
14:         break;
15:       end if
16:     else
17:       break;
18:     end if
19:   end for
20:   if Flag = false then
21:     push p into M;                        # if not occluded, push into patch data
22:   end if
23: end for
```

3.3 VIEWPOINT ORIENTED COLOR-SHAPE HISTOGRAM

For recognition and pose recovery of everyday objects, use of an object descriptor which consists of both color and shape information is a prerequisite. In particular, this descriptor needs to be able to differentiate the objects which have the same shape but different colors and also deal with both textured and textureless objects. In order to fulfill the aforementioned requirements, a novel object descriptor called *viewpoint oriented color-shape histogram* is proposed here based on both color and shape features. During VCSH construction, firstly the color of each point p in object patch data M_j is smoothly ranged and color distributions for different ranges are estimated. Secondly,

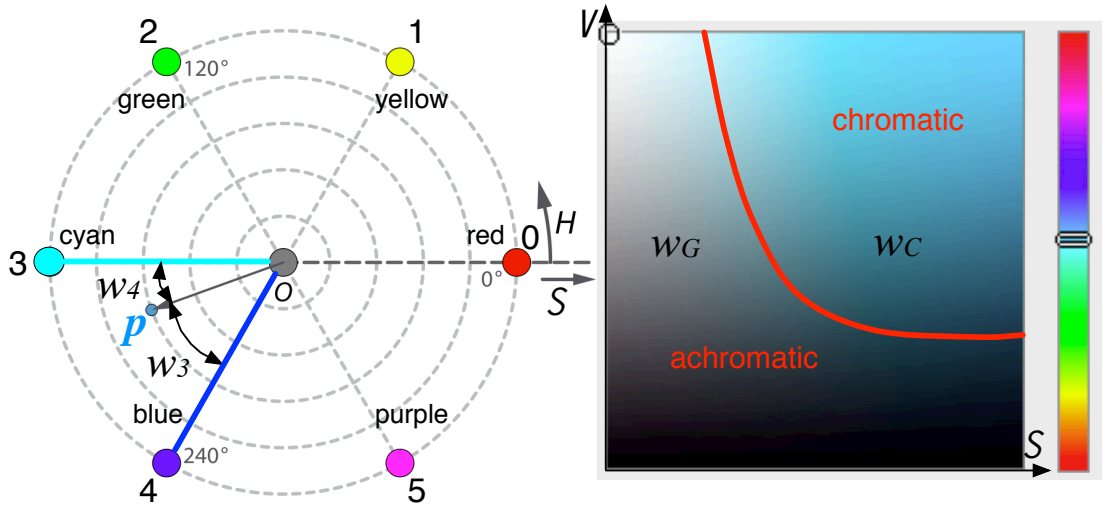


Figure 3.3: Left: smoothed color range and estimation of contributions for adjacent regions in HS space. Right: illustration for the chromatic and achromatic areas in SV space.

the shape features are estimated to describe the geometrical relationship of each point with the viewpoint VP_j and the M_j's centroid c. Finally, the extracted color and shape features are correlated and built as a VCSH to describe each object patch data M_j.

3.3.1 *Smoothed Color Ranging*

To represent the uniqueness of a color feature for all object patch data, the feature needs to be characterized and color distributions for different ranges need to be estimated according to their color values. The HSV color space is employed here for better characterizing the color features of each point due to its robustness to illumination changes [48]. As shown in Figure 3.3, there are chromatic and achromatic areas in SV space, in which the chromatic area represents the true color space while achromatic area represents the gray scale space. That is, the histogram is divided into 8 regions as RE_u with the index of $u = \{0 \cdots 7\}$, in which six are for the chromatic area, and the other two are for the achromatic area [142].

In more detail, firstly, we consider the six true color histogram regions RE_0 to RE_5, which represent six typical colors CR_0 to CR_5. Each point's hue value can then be quantized into a certain color region CR. However, the hard quantization cannot represent the true color correctly. To overcome this issue, a smoothed ranging method is proposed, by estimating two distributions w_H for two consecutive histogram regions RE in true color space. The detailed steps are presented as follows:

- Identify CR_n: red as $CR_0 = 0$, yellow as $CR_1 = 60$, green as $CR_2 = 120$, cyan as $CR_3 = 180$, blue as $CR_4 = 240$, purple as $CR_5 = 300$. Consequently, six

histogram ranges are divided based on the color index CR, as $RE_u \rightarrow CR_n$ where $u = n = \{0 \cdots 5\}$.

- For each color point p, its hue value H is ranged into two consecutive histogram regions RE_u and RE_{u+1} as $u = \lfloor H/60 \rfloor$, if $u = 5$, the next histogram region RE_{u+1} would be reset to RE_0.
- Estimate color distributions w_{H_u}, $w_{H_{u+1}}$ according to the ranged adjacent regions RE_u, RE_{u+1} in true color space, based on the distance from hue value H to CR_n and CR_{n+1}:

$$\begin{aligned} w_{H_u} &= (H - CR_{n+1})/60, \\ w_{H_{u+1}} &= 1 - w_{H_u}. \end{aligned} \tag{3.2}$$

Secondly, we consider the achromatic area which consists of two histogram regions RE_6 and RE_7. When one of the saturation S and value V is near zero in HSV space, the point color will be represented in gray scale. Since the color in achromatic space is highly sensitive to illumination changes, the previous estimated distributions w_{H_n} and $w_{H_{n+1}}$ in true color space needs to be redesigned according to the influence from S and V. In order to capture the nature color, a soft decision method [155] is employed and we update both chromatic and achromatic components of the histogram. The weight w_C of the chromatic and w_G of achromatic components is determined by S, V, and their sum equals unity:

$$\begin{aligned} w_C &= S^{r(1/V)^{r_1}}, \\ w_G &= 1 - w_C, \end{aligned} \tag{3.3}$$

where $r, r_1 \in [0, 1]$. The latter are empirically chosen to be $r = 0.14$ and $r_1 = 0.9$ to give the best precision on true color. Furthermore, the value of saturation V is quantized. Based on V, the distributions w_6 and w_7 are calculated for regions RE_6 and RE_7: $w_6 = w_G$ if $V < 0.5$, otherwise $w_6 = 0$; while the value of w_7 is the reverse.

We therefore update all the previous estimated color distributions as w_u and w_{u+1}, by considering the chromatic weight w_C's influence on true color representation.

$$\begin{aligned} w_u &= w_{H_u} \cdot w_C, \\ w_{u+1} &= w_{H_{u+1}} \cdot w_C. \end{aligned} \tag{3.4}$$

Finally, each point p with HSV color value is ranged into three histogram regions $\langle RE_u, RE_{u+1}, RE_6 | RE_7 \rangle$ with respective contributions being $\langle w_u, w_{u+1}, w_6 | w_7 \rangle$.

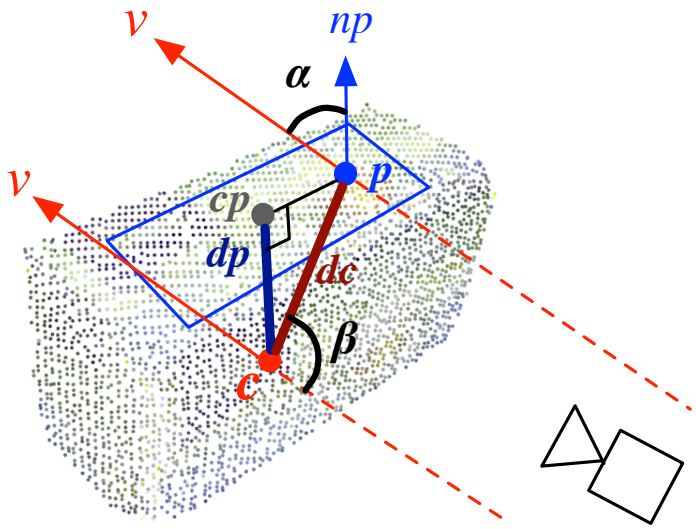


Figure 3.4: Shape features of point p. c is the centroid of object patch data. n_p is the normal of p. v is the synthetic viewpoint direction. c_p is c's projection point on the tangent plane of p (blue rectangle frame). d_c and d_p are the distances from c to p and from c to c_p. α is the angle between v and n_p, and β is the angle between v and the line segment cp.

3.3.2 *Shape Feature Extraction*

After the color contributions have been estimated for the specific histogram regions, it is necessary to extract each object patch data M's shape features $F = \{f_0 \cdots f_m\}$ for the final histogram building, where m is the point number in M. With object patch data M representing the partial data of the object from viewpoint VP, the geometrical information of point p can be extracted in order to describe the object shape accurately and robustly. Partly inspired by the work in [5], we extract the shape features depending on point p's relationship with the centroid of M and viewpoint VP. As a global descriptor, the surface normal n_p of each point p in M and the centroid c of M are computed at first. The relationship of p and c represents the 3D shape of the object cluster. The relationship of p and VP indicates the rotation of the object cluster relative to the sensor direction. Note that VP and c represent the object's 6D pose.

As shown in Figure 3.4, the tangent plane of p is defined as a plane that is orthogonal to p's normal n_p. The centroid c is projected on this tangent plane as a point c_p.

A four dimensional geometrical feature f consists of two distances and two angles components $\langle d_p, d_c, \alpha, \beta \rangle$, which are calculated as:

$$\begin{aligned} d_p &= \|p - c\|, \\ d_c &= \|c_p - c\|, \\ \alpha &= \arccos(n_p \cdot (p - c)), \\ \beta &= \arccos(v \cdot (p - c)). \end{aligned} \tag{3.5}$$

In the object partial data M, the feature f is calculated for each point p. Therefore, for a single object model O which contains J object patch data, the final feature set is $F = \{f_0 \cdots f_m\}$ with m points, representing the object's shape from a certain viewpoint VP_j.

3.3.3 *Color and Shape Feature Correlation*

VCSH descriptor needs to be correlated with color and shape features to describe an object's patch data M with the viewpoint VP discriminatively and comprehensively as a histogram. In the smoothed color ranging phase, the entire histogram are combined with eight regions. Every component in each point's shape feature f has 30 bins, therefore each RE contains 120 bins inside. Each p's two distance components $\langle d_p, d_c \rangle$ are indexed as $\langle IN_{d_p}, IN_{d_c} \rangle$ by the quantization using their values scaling from minimum value $\langle d_{p_{min}}, d_{c_{min}} \rangle$ to maximum value $\langle d_{p_{max}}, d_{c_{max}} \rangle$. Two angle components $\langle \alpha, \beta \rangle$ are indexed as $\langle IN_\alpha, IN_\beta \rangle$ by the quantization using their values with the range of 0 to 90°as follows:

$$\begin{aligned} IN_{d_p} &= \lfloor \frac{30 \cdot (d_p - d_{p_{min}})}{d_{p_{max}} - d_{p_{min}}} \rfloor, \\ IN_{d_c} &= \lfloor \frac{30 \cdot (d_c - d_{c_{min}})}{d_{c_{max}} - d_{c_{min}}} \rfloor, \\ IN_\alpha &= \lfloor \frac{\alpha}{90} \cdot 30 \rfloor, \\ IN_\beta &= \lfloor \frac{\beta}{90} \cdot 30 \rfloor. \end{aligned} \tag{3.6}$$

During the object's color and shape features correlation step, each p's color contributions as $\langle w_u, w_{u+1}, w_6 | w_7 \rangle$ for three histogram regions $\langle RE_u, RE_{u+1}, RE_6 | RE_7 \rangle$ are incrementally added into $\langle INX_{d_p}, INX_{d_c}, INX_\alpha, INX_\beta \rangle$. The final certain bins index

INX in VCSH regarding to each of these three $RE_m, m \in [u, u+1, 6, 7]$ are quantized as follows:

$$\begin{aligned} INX_{d_p} &= IN_{d_p} + 120 \cdot m, \\ INX_{d_c} &= IN_{d_c} + 120 \cdot m + 30, \\ INX_{\alpha} &= IN_{\alpha} + 120 \cdot m + 30 \cdot 2, \\ INX_{\beta} &= IN_{\beta} + 120 \cdot m + 30 \cdot 3. \end{aligned} \tag{3.7}$$

The entire histogram has incremental values corresponding to color contributions from all the points in M. During the final object recognition phase, the object's descriptor should not change with varying distance but constant view direction. However the histogram's absolute value of each bin will change according to the object cluster point number. To overcome this problem, the values of the histogram are finally normalized using the point number. Thus, the VCSH can be viewed as a geometrical constrained color feature histogram. As shown in Figure 3.5, color contributions of all points in object patch data respected to different viewpoints are incrementally added into the certain indexes of whole VCSH, based on smoothed color ranging and shape feature extraction. An example of two picked points in object patch data for the final VCSH generation is illustrated in Figure 3.5, with the step of color-shape features extraction and correlation step. The patch data of object can then be represented as one VCSH. The final correlated histogram has $(6+2) \times (30 \times 4) = 960$ dimensions. The computational complexity of VCSH is $O(n)$, where n is the point number of object patch data M. Consequently, the final generated histogram gives the possibility of achieving a high object recognition rate as well as accurate pose estimation while maintaining real-time processing capabilities.

3.4 MULTIPLE OBJECT RECOGNITION AND POSE RETRIEVAL

3.4.1 *Object Recognition and Initial Pose Estimation*

We are now going to get the identification label L and the general pose P of the object cluster in the real scene using the VCSH descriptors dataset. Our system first segments and clusters the object cluster C from the background. Two frameworks of segmentation and clustering are proposed to accommodate different environments for object recognition and pose estimation:

Planar Background Environment The environment can be simplified when all the objects share a planar background, for example a table surface as shown in Figure 3.9a. Utilizing the raw RGB-D image from a Kinect sensor, the largest plane surface can be

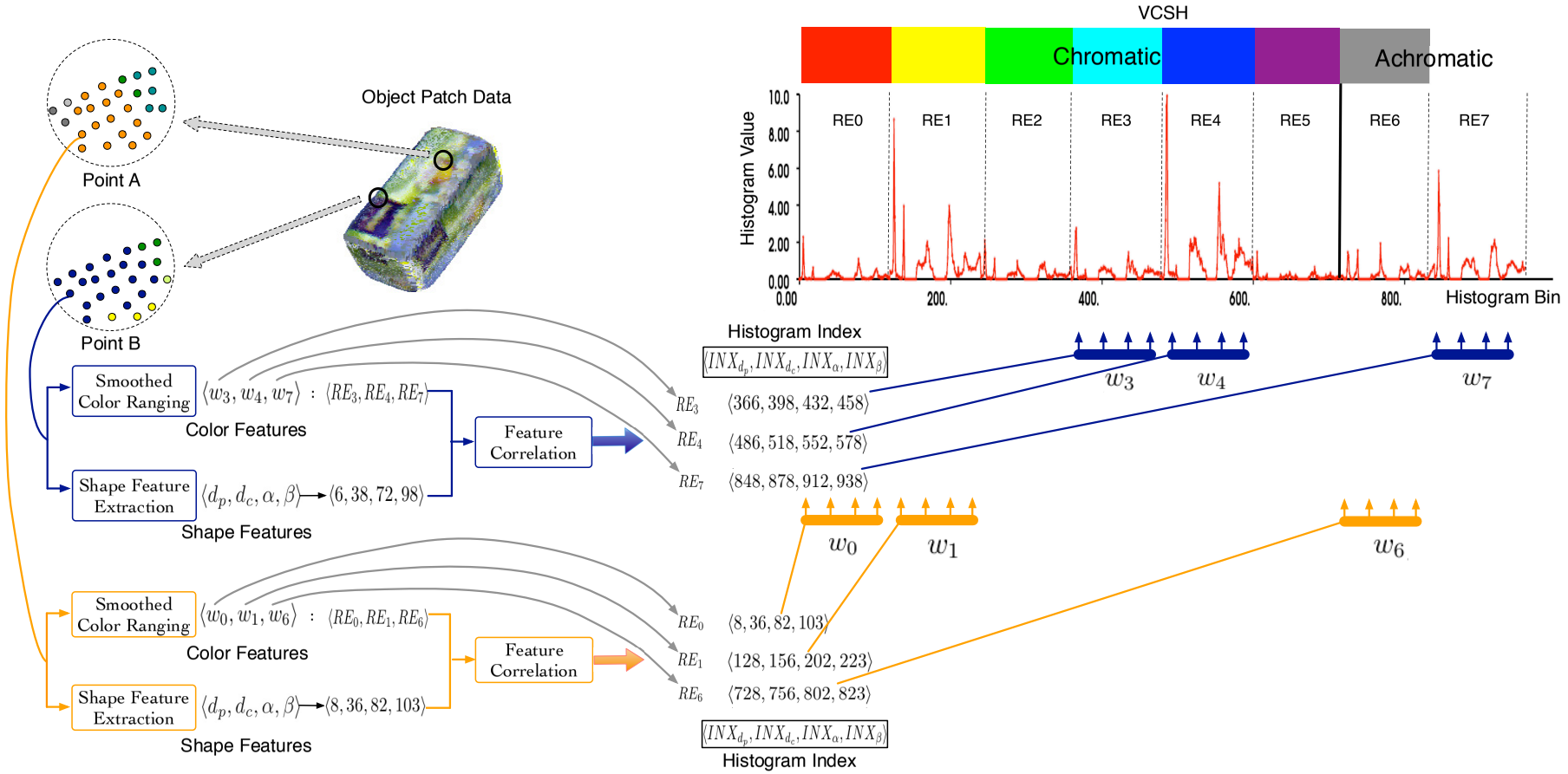


Figure 3.5: VCSH training process featuring smooth color ranging, shape feature extraction and feature correlation steps. The color contribution of all the points in the object patch data with respect to different viewpoints is indexed in the VCSH based on smoothed color ranging and geometrical features.

extracted by RANSAC [119]. The object clusters C_k are then segmented from the plane surface and clustered by Euclidean distance [163].

Cluttered Background Environment The cluttered background environment is represented as a heavily cluttered background. It is difficult to constrain the objects' localization for segmentation and clustering, as the target objects have the possibilities of being with various pose as shown in Figure 3.9b. Aiming to solve that, the initial background image is trained in off-line phase based on Octree data structure [36]. With the extracted foreground data, the object clusters C_k will be segmented and clustered by Euclidean distance [160].

Based on object clusters C_k, the real scene objects' VCSH is calculated. The chi-squared distance χ^2 between the real scene object's VCSH value $H(C)$ and H_{ij} in the trained dataset is calculated for the best matching, through fast approximate K-Nearest Neighbors (KNN) method based on kd-trees [119]. $\langle \hat{L}, \hat{P} \rangle$ as the best matched object identification and corresponding pose can be extracted as:

$$\langle \hat{L}, \hat{P} \rangle = \underset{\langle L,P \rangle_{ij}}{\arg\min}\, \chi^2(H(C), H_{ij}). \tag{3.8}$$

Note that in VCSH definition, P in $\langle \hat{L}, \hat{P} \rangle$ represents the rotation of the object respect to the sensor's viewpoint. The centroid of the object cluster in real scene indicates the current position, which is used to update P as the object initial pose.

3.4.2 *Object Pose Optimization and Verification*

Even though the estimated pose P is recovered as the best matched pose from the model dataset, P may be not the real pose. This is due to the sampling rate of the synthetic viewpoints during the generation of the VCSH dataset. Consequently, the iterative closest point (ICP) method is employed to further optimize the estimated pose [176, 7], providing a transform T_{icp}. The sources for the ICP are the point cloud data of the best matched object patch data and the object cluster in real scene. The accuracy and speed of the ICP strongly rely on the given initial guess, which can be provided by our estimated pose P. The final pose of the object P_{final} is optimized according to the extracted initial pose P and the ICP optimized transform T_{icp}. Therefore, the accuracy of the final object pose $P_{final} = T_{icp}^{-1} \cdot P$ can be significantly improved. Moreover, the iteration speed is high enough for real-time recognition and pose estimation.

A pose verification step is necessary to make sure that the optimized pose P_{final} is the correct estimation. The new object patch data M_{rec} will be generated by P_{final} and the 3D model Ω of the recognized object using Algorithm 3.1. Since the final pose

is optimized, the detected object patch data M_{rec} might be not in the object model patch dataset that generated from the synthetic viewpoints during modeling. False positives can be rejected using the difference between M_{rec} and the real object cluster data C_k with appropriate thresholds for photometric and geometric differences.

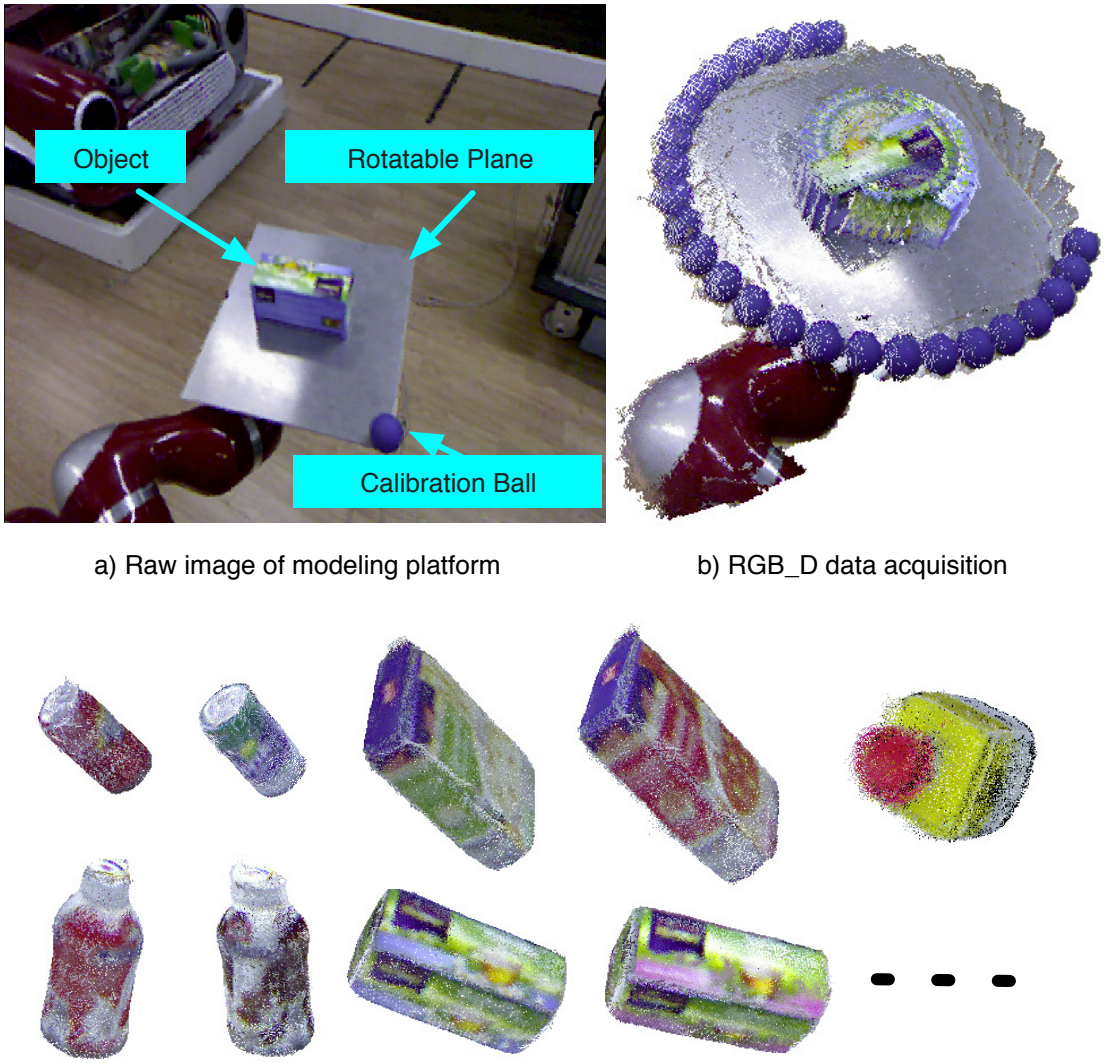
3.4.3 *Object Localization in a Large-scale Semantic Map*

Semantic mapping has attracted huge attention in robotic applications, especially for wide-range navigation and exploration. Therefore, it is obvious that a coherent semantic map, which provides both semantic level understanding and metric representation of the environment, is very important for an intelligent robot to successfully and efficiently perform daily tasks. To fulfill these requirements, VCSH is an important component in building a coherent semantic map. Specifically, VCSH allows the localization of objects in large-scale semantic maps through its efficient and accurate 3D object recognition and pose estimation. This can be achieved in real time while the robot is building the map. Moreover, VCSH imposes no constraints with respect to the object type and can deal with both textured and textureless objects using color and shape information.

We employ a two-step coherent map building strategy. In the first step, laser range data is processed by a grid mapping algorithm, in this case GMapping [52]. This results in an occupancy grid map of the environment and provides a coherent global coordinate system [152]. The resulting grid map is then used as input for the process of parametric environment abstraction which uses rectangular space units to approximate the geometry and the topology of the perceived environment [89]. Within each space unit, unknown areas of the grid map are detected using connected-components analysis [23]. Such areas are considered to be obstacles which cannot be traversed by robots [57]. More details on parametric environment abstraction can be found in [90]. 3D objects are localized in the global semantic map using our proposed object recognition and 6D pose estimation method. Finally, a coherent semantic map that captures the geometrical, topological and object information of the operating environment is generated by incorporating the 3D object information into the parametric environment model.

3.5 EXPERIMENTAL RESULTS

We performed experiments where the goal was to evaluate our proposed *viewpoint oriented color-shape histogram* descriptor as well as the entire object recognition system. At first, an object dataset consisting of more than 25 objects is built, where some



a) Raw image of modeling platform

b) RGB_D data acquisition

c) Some samples of the built 3D object models

Figure 3.6: Object 3D modeling where model data are represented as color point clouds. a) the platform used for obtaining object models; b) captured object data using the Kinect sensor; c) a selection of objects models in our dataset.

objects have the same shape but differ in surface color information. As shown in Figure 3.6a, a platform was developed that can rotate by different angles using a KUKA arm end-effector controller. With a stationary Kinect sensor mounted on the robot, the color point cloud of the object can be captured under different angles of rotation corresponding to different object poses. Furthermore, a calibration ball is used to determine and optimize the coordinate system of the final object model. In total, for each object, 25 frames of data using 10° as an angle step are captured. Some objects have the same shape but different color information such as the COLA and SPRITE cans. Some objects are textureless such as the emergency button (Figure 3.6c). Due to the way the data is captured, the part of the object that is in contact with the platform is not considered in the object model.

Figure 3.7: Other object modeling platform. These object models can also be successfully applied in our system.

During the object patch data generation, the viewpoints are sampled on the upper sphere surface around the object origin with radius of 0.8 m. A synthetic viewpoint and the corresponding object patch data are generated for all combinations of elevation θ_e and azimuth θ_a, where $\theta_e \in [10°, 80°]$ with a step of 10° and $\theta_a \in [0°, 360°]$ with a step of 2°. Therefore, $7 \times 180 = 1260$ synthetic views patch data for each object model are generated in total. In our dataset, each viewpoint object patch data contains around 1000 to 2000 color points. Consequently, each object is represented as 1260 VCSH descriptors. Each of these descriptors represents a different viewpoints, covering the full range of potential object poses. Note that we also tested other object modeling strategies including Kinect Fusion [69], CAD modeling and public datasets [74, 83] as shown in Figure 3.7. By converting these different object model types into colored point clouds, these objects can be successfully modeled by our system.

To demonstrate its performance, we designed multiple challenging scenarios. Some special objects are chosen to present VCSH's stability of recognition and also pose accuracy. We used objects which have the same shape but different visual information, as well as objects with textured and textureless surface. High-speed recognition and accurate pose estimation of common objects remains a challenge that has not been satisfactorily solved by existing techniques [27, 73, 74, 118, 119, 146].

At first, we validate the effectivity and efficiency of our proposed VCSH descriptor using offline simulations. The selected object patch data in our dataset is chosen to be

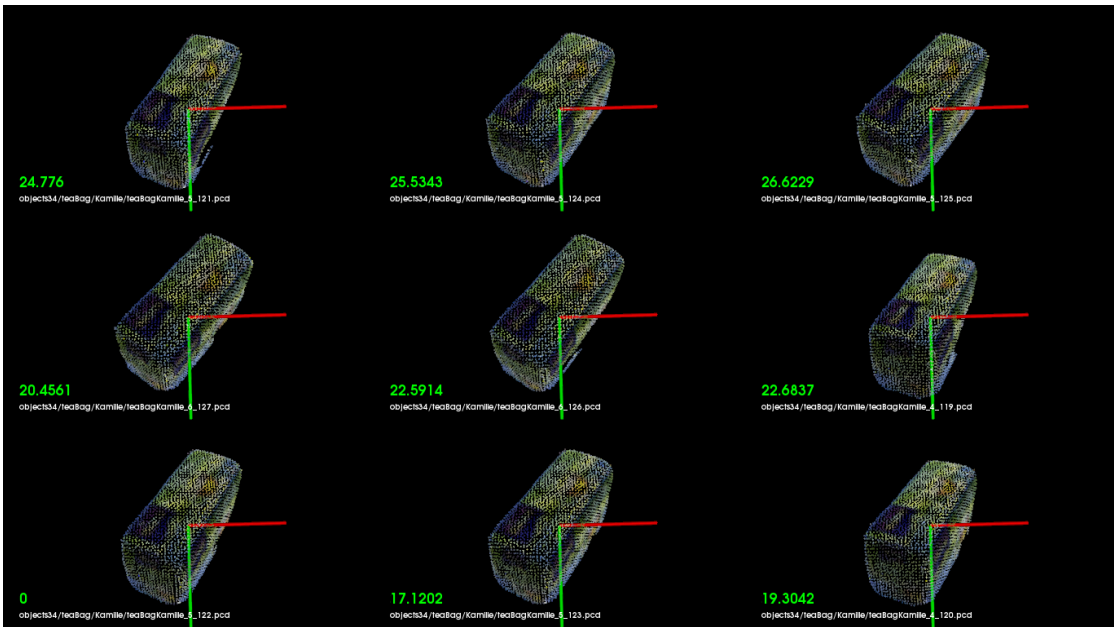
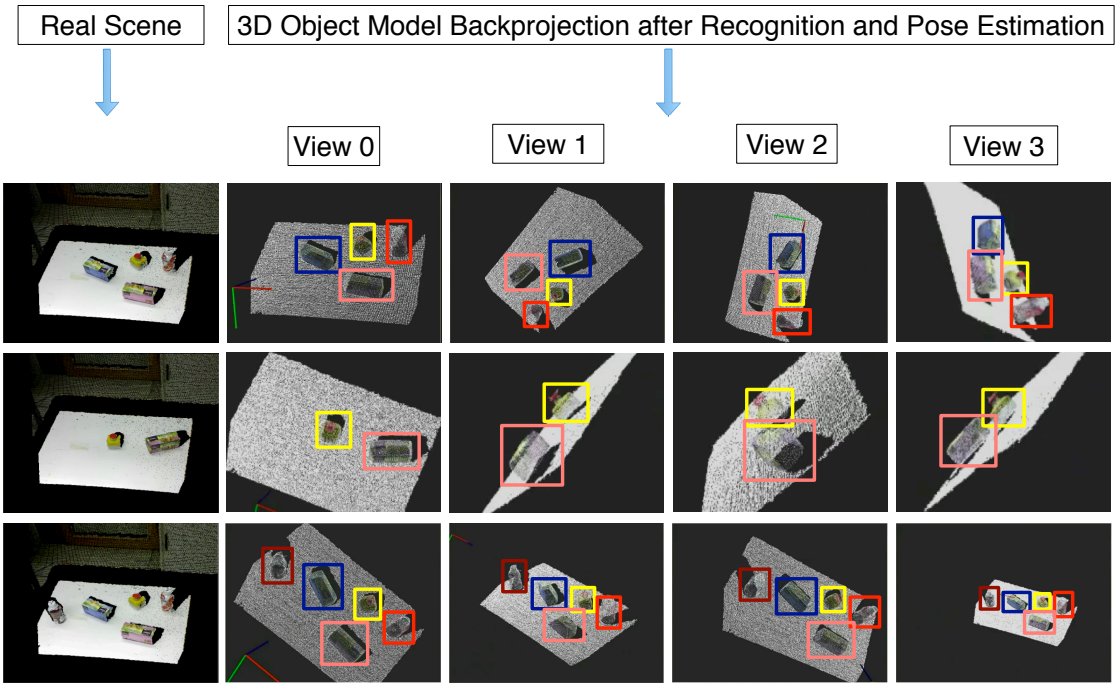


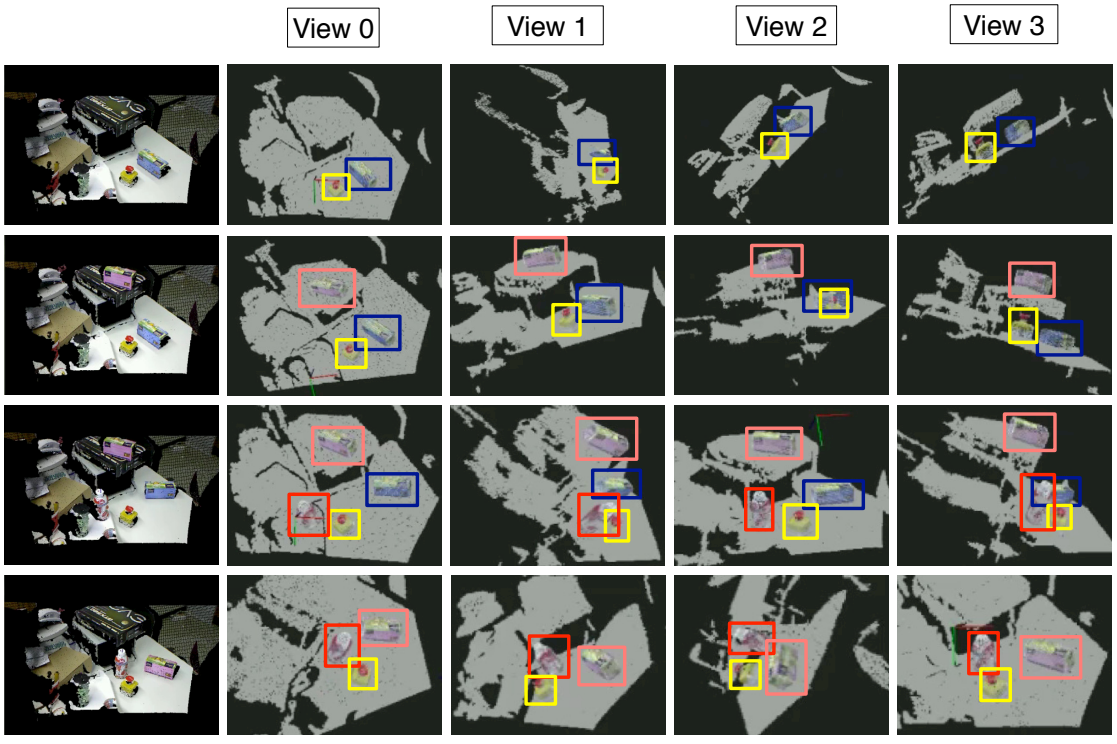
Figure 3.8: Extract the nine closest object VCSHs with relevant viewpoints in the dataset (after the recognition step using the simulation data). The green markers present the distances to the chosen target VCSH.

generated as a VCSH. Then it is used to retrieve the closest VCSH as the best matched to evaluate this accuracy. As shown in Figure 3.8, the best matched VCSH in dataset is estimated to present the correction of its recognition and the relevant viewpoint retrieval. The green scores present the nine closest VCSH by comparing its histogram distance respect to the target VCSH. The object patch data for test can be recognized correctly and its relevant pose can be reached. From our testing, all of the object patch data in our dataset can be correctly matched with 100% success rate.

We now demonstrate real time object recognition and 6D pose estimation using the real scene RGB-D data, which is captured from a single Kinect on an autonomous mobile robot. Because of the data acquisition range of the Kinect, the objects need to be within a distance of 0.5 m to 3.5 m with respect to the sensor. The 3D models of the recognized objects are projected into the real scene with the estimated 6D poses as shown in Figure 3.9. For the planar background scenario shown in Figure 3.9a, we extracted the object cluster under the assumption that all the objects stand on a planar surface. The latter needs to cover at least 50% of the point cloud points captured by the sensor. Figure 3.9b illustrates the cluttered background scenario. The background is necessary to be trained at first, and all the objects have no geometrical constrains in real scene. The objects for the experiments include the textured (tea box and milk bottle) and also the textureless (emergency button). Same shape and different color objects are also tested such as various tea boxes to present the necessary for the object descriptor combined with color and shape features. All the trained objects can be



a) Experiment Results in Planar BackGround Environment



b) Experiment Results in Cluttered Background Environment

Figure 3.9: 3D models of recognized objects are projected onto the real scene with estimated 6D poses: a) using a planar background environment; b) using a cluttered background environment. (left column) RGB-D data from the real scene. (columns 2-5) the different view results after the object model is backprojected into the scene data after recognition and pose estimation. Different color frames illustrate different objects.

correctly recognized and their estimated poses are highly accurate. Note that the Point Cloud Library[1] was used in obtaining these results.

Our VCSH is a global object descriptor combining color and shape features. The later is based on the centroid of the object cluster in the real scene. Due to its nature, use of the VCSH imposes certain limitations. Specifically all the objects must be rigid and must not be reflective or transparent. In addition, these objects should be well segmented from their background environment. As described in Section 3.4 and demonstrated by the experimental results shown in Figure 3.9, our system can effectively deal with both planar backgrounds and cluttered environments. To analyze the influence of object occlusions on the final results, we utilized multiple experiments for multiple objects with manual configurations for occlusion. During the experimental testing, if the occluded colored point clouds are less than 8% of the ideal whole object data, our VCSH provides stable and correct results for both recognition and 6D pose estimation.

Furthermore, we apply our object recognition and 6D pose estimation method in semantic mapping of an indoor environment, as illustrated in Figure 3.10. As shown in Figure 3.10a, the resulting coherent semantic map correctly interprets the perceived environment with space units $U1, U2 \cdots U6$ and their corresponding topology (connectivity by doors and adjacency). In Figure 3.10b, the detected obstacles represent the furniture of the perceived environment, such as tables and cabinets. 3D parametric models along with the detected 3D objects are shown in Figure 3.10c. Here the detected table planes and objects are back-projected in the map. Figure 3.10d depicts the details of object recognition and localization. In space units $U1, U2$ and $U5$, several 3D objects are recognized and localized with respect to their 6D poses. By cell-wise checking of our parametric model and the input grid map, we measured an accuracy of 94.1% in geometry approximation. The mismatch of 5.9% is mainly due to some not-fully-explored areas of the input map.

Table 3.1 presents the state-of-the-art methods on the topic of object recognition and 6D pose estimation. There are mainly two types of descriptors including global and local. In particular, the local type is similar to the method of model registration. It can solve the problem when object data contains occlusions using the pairwise matching of different features. However, this method incurs a high computational cost and is not suitable for real-time processing such as robotic applications. Furthermore, most of the local object descriptors must have the prior knowledge about the existence of the object in the real scene (see CPPF [27]). Instead, in this work, we introduce a new global object descriptor VCSH. Compared with other global methods like VFH [119] and Tang [146], we can retrieve accurate 6D pose, which cannot be solved in VFH. Moreover, VFH only uses shape features and as a result it cannot distinguish between objects of the

1 http://www.pointclouds.org

Table 3.1: Map of the state-of-the-art methods on 3D object recognition and pose estimation: ConVOSH, CPPF and our VCSH can be applied for textured and textureless objects. ConVOSH cannot retrieve 6D pose. CPPF as a local descriptor has a high computational cost that precludes real-time application. Notice that the numerical values come from their respective papers, and in particular those numbers refer to their own datasets, therefore the results can only provide a rough comparison.

Name	Strategy	Type	Feature	Object	Dataset	Success	Pose Error	
	Recognition(R) Registration(RG) 6D Pose(P)	Local(L) Global(G)	Shape(S) Texture(T) Color(C)	Constrains	Size	Rate (%)	T (mm)	R (deg)
ConVOSCH[74]	R	L	S + C	No	63	72.2	N/A	N/A
LINEMOD[62]	R+P	L	S + C	Uniform Color	15	96.6	N/A	N/A
VFH[119]	R	G	S	Depth Only	60	98.1	N/A	N/A
CPPF[27]	RG +P	L	S + C	No	10	80	15	15
Tang[146]	R+P	G	S + T	Textured	35	90	50	10
Our VCSH	R+P	G	S + C	No	25	92	23.4	1.59

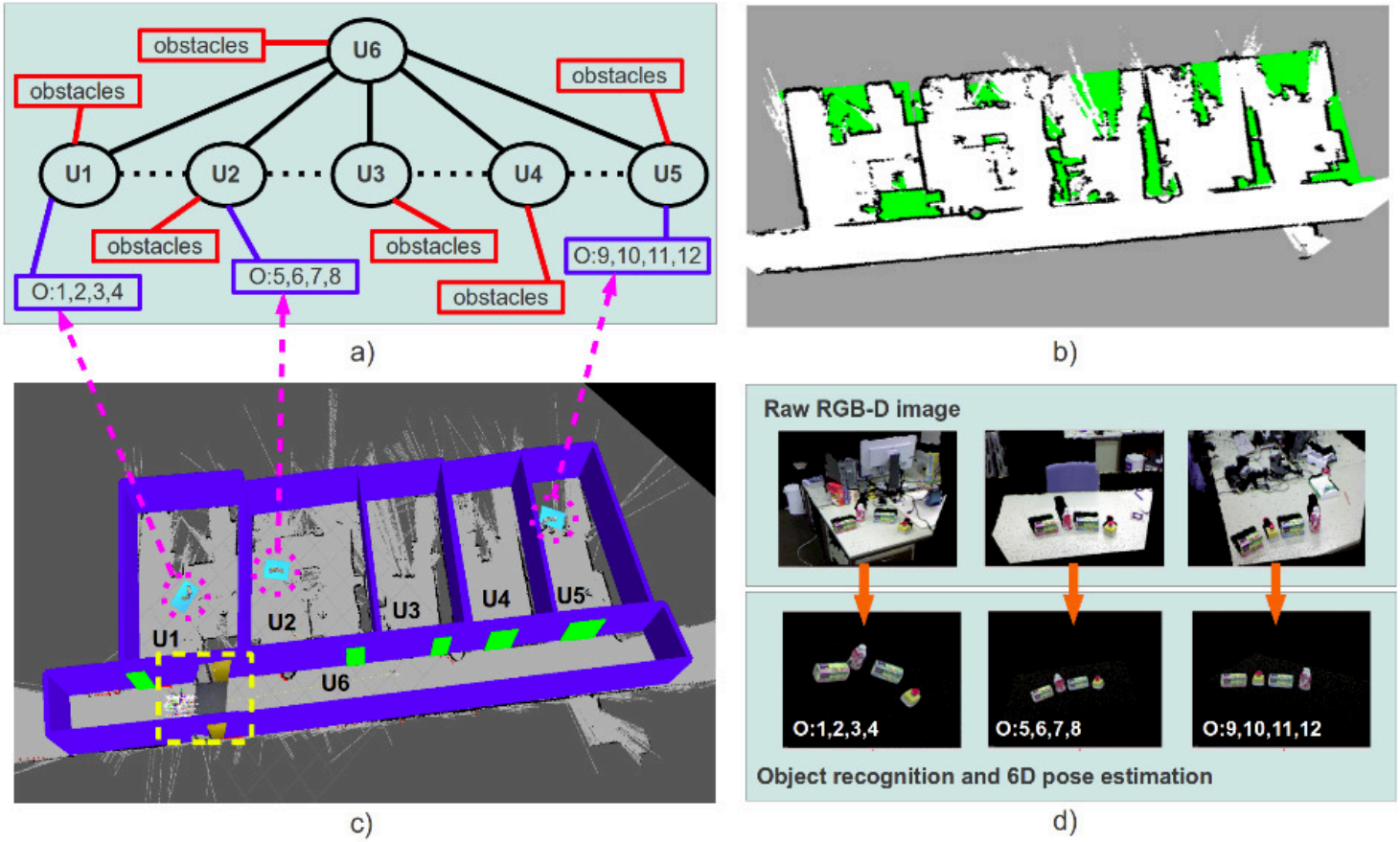
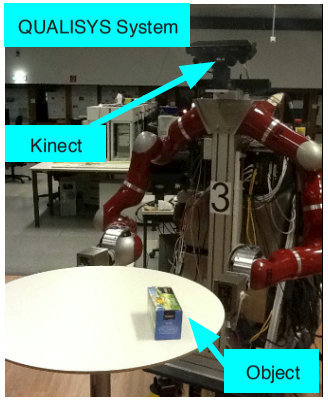
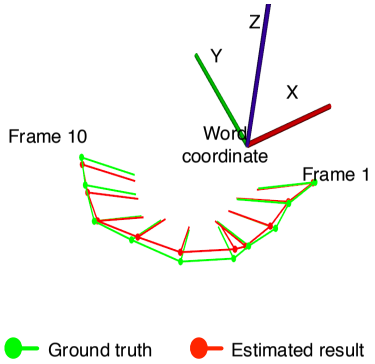


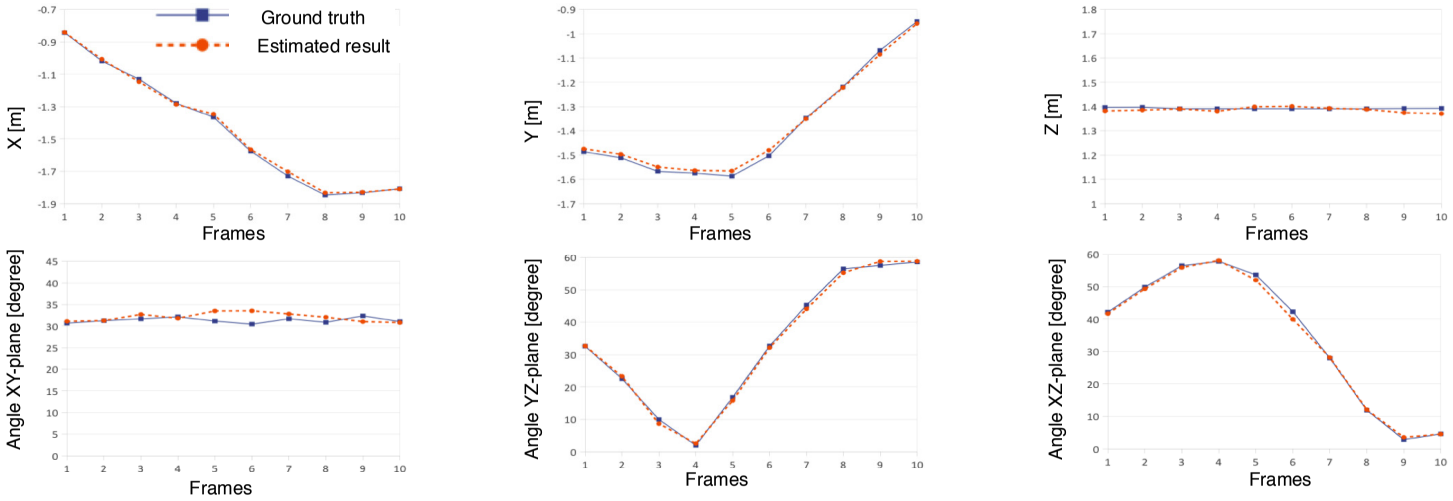
Figure 3.10: Object localization in coherent semantic maps. a) The abstract environment model. Black ellipses indicate space units. Solid black edges mean that two space units are connected by one or more doors. Dashed edges imply that two space units are adjacent but not connected by doors. Blue rectangles show the detected objects. Blue edges show the belongingness of these objects. b) The resulting grid map of the perceived environment. c) We plot the 3D semantic map directly onto the corresponding grid map (blue=walls, green=doors, cyan=detected tables with 3D objects). The current robot information including acquired RGB-D data are highlighted by the dashed yellow rectangle. d) Details on 3D object localization. This semantic map includes the identification and pose of each object in the global coordinate system and where they are located in the semantic map.



a) Experiment setup for pose accuracy evaluation



b) Estimated pose and ground truth of the sensor



c) 6D pose estimation in different frames (translation and rotation)

Figure 3.11: Object pose accuracy evaluation in different frames with different robot positions: a) experiment setup for evaluation with omni-direction platform robot and QUALISYS tracker system; b) the estimated sensor trajectory with 10 frames; c) the estimated pose and ground truth in translation and rotation.

Table 3.2: Runtime performances of our VCSH and Tang [146] on similar scenarios.

Single Object	Train	Feature Extract	Recognize	Pose Recovery
Our VCSH	2 min	5 ms	37 ms	0.83 s
Tang [146]	7 min	5 s	1 s	14 s

same shape but different visual appearance. Tang [146] uses the SIFT feature based on surface texture information. This method requires that the target objects are textured with high quality and cannot deal with textureless object, such as the "emergency button" in our dataset. Our VCSH is based on color and shape features and thus has no object model type constraints dealing equally well with extured and textureless objects. Comparatively, most other methods impose constraints on the types of object models, such as textured (Tang [146]), uniform color (LINEMOD [62]), depth only (VFH [119]). To the best of our knowledge, we are currently among the first to solve these problems with high recognition rate, accurate 6D pose estimation and low computational cost for any type of object by combining the photometric and geometric information.

After 1000 demonstrations our framework was able to correctly recognize the object and determine its pose in 92% of the cases. The object was correctly recognized but its pose was wrong in 6% of the cases and only 2% of the cases resulted in wrong recognition. Achieving a good runtime performance is very important for applying our framework into applications involving autonomous mobile robots. The runtime performance for single object recognition and pose recovery are evaluated and compared with the results from Tang et. al. [146] for similar setups. Results are shown in Table 3.2. All experiments ran on AMD X6 3.0 GHz with 8 GB of RAM, while Tang et al. used a 6-core 3.2 GHz i7 with 24 GB of RAM. One second was required for single object recognition and pose recovery without employing any GPU speed-ups. This runtime performance enables implementation in real time robotic applications, for instance, object grasping and manipulation based on an appropriately designed perceptual system.

To further evaluate the pose accuracy using our proposed approach, the QUALISYS motion capture system[2] is employed to capture the ground truth of the sensor pose while the robot with the Kinect senor moves around the stationary object. The camera pose is estimated with two methods for accuracy analysis: 1) recovered pose respect to the stationary object from our proposed method; 2) estimated pose using QUALISYS system as the ground truth. By transforming these data into the world coordinate system, we can compare the estimated pose with its ground truth to get the pose

2 http://www.qualisys.com/

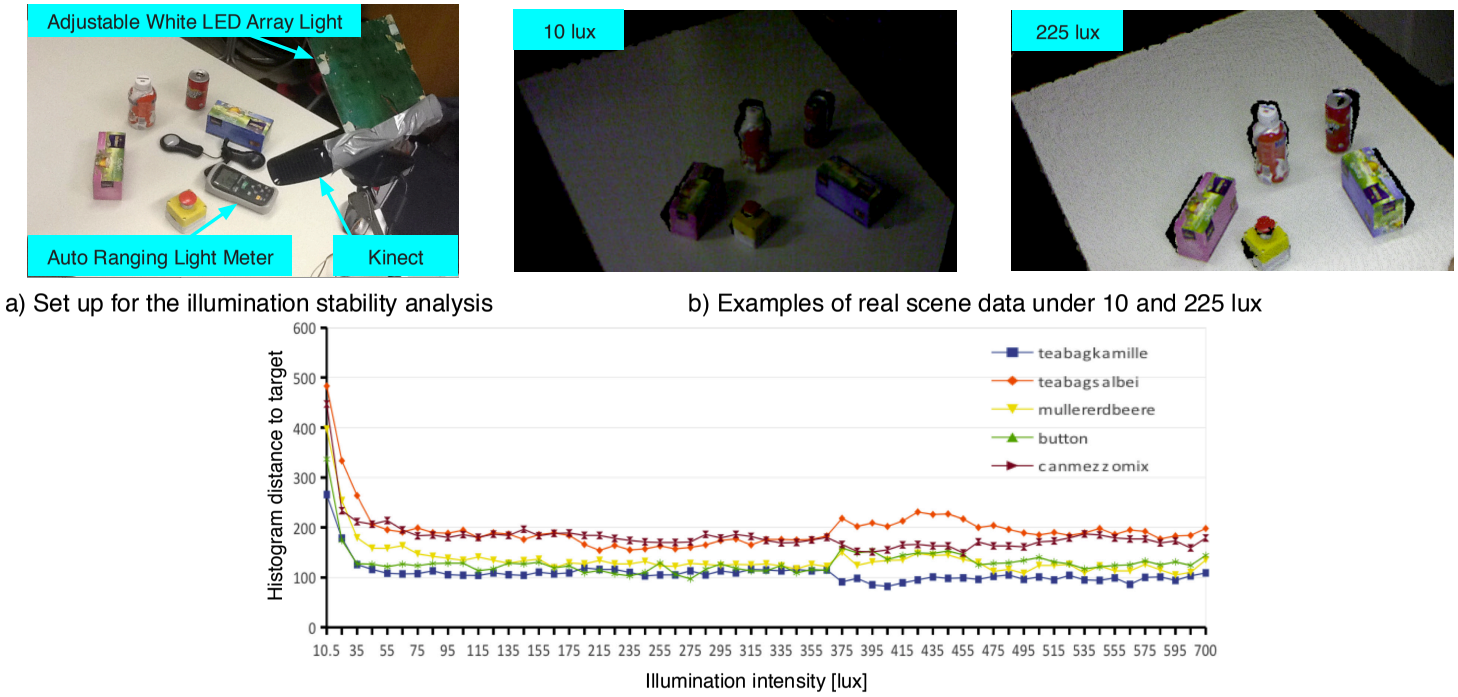


Figure 3.12: Stability analysis with illumination change: a) experimental setup with adjustable white LED array and light meter to measure the illumination density; b) some real scene data recored under the different illumination densities; c) five objects's estimated VCSH distances to the relative targets under sixty different illumination densities from 10 lux to 700 lux.

recovery accuracy, as shown in Figure 3.11. The root mean square error (RMSE) during the whole 10 frames is calculated for the pose accuracy analysis. From Table 3.1, the 6D pose error compared with the ground truth is 23.4 mm in translation and 1.59 degrees in rotation, while in [146] it amounted to 50mm and 10 degrees respectively. Our VCSH outperforms Tang et. al.'s method both in translation and rotation accuracy with similar object models for similar scenarios.

Since color information is used as a photometric feature during the generation of the VCSH, its stability with respect to illumination changes is a crucial requirement that needs to be analyzed. We utilize one light meter DT1309 to estimate the illumination intensity around the object under an adjustable white LED array light. The stability is evaluated by the difference between the estimated VCSH under various illumination conditions and the target VCSH (correct identification and pose) captured in the dataset. As illustrated in Figure 3.12, even when the illumination intensity exceeds 50 lux, all the histogram differences remain under 220 and the VCSH is stable until 700 lux, which is the maximum illumination intensity in a daily environment. Note that the object modeling environment is approximately 230 lux, while most of the common indoor and outdoor light conditions range between 150 and 400 lux. From the result of the stability analysis, our recognition and pose estimation framework and in particular the VCSH object descriptor is stable enough under varying illumination.

Based on our experimental results, we conclude that our proposed approach consisting of a novel object descriptor VCSH is efficient and robust. It guarantees high object recognition rate, fast and accurate pose estimation as well as exhibits the capability of dealing with illumination changes.

3.6 SUMMARY

In this chapter, we presented a framework consisting of a global object descriptor *Viewpoint oriented Color-Shape Histogram*, which combines color and shape information for object recognition and 6D pose estimation. The proposed approach can be easily integrated into various robotic perception systems for textured/textureless object recognition and 6D pose estimation in real time. In addition, we successfully incorporated this approach in a coherent semantic map, which can be used for robot exploration of objects in large-scale map.

Our approach achieves 92% success object recognition rate for both of correct object identification and pose retrieval. The estimation error of the 6D pose is under 24mm in translation and 1.6 degree in rotation. Our proposed framework also has a low computation cost. For a single object, it requires less than 1 s to recognize and accurately estimate it's pose after pose optimization. Moreover, our VCSH is efficient and stable

enough under varying illumination conditions found in common environments. Our experimental results demonstrate that the proposed approach is efficient by guaranteeing high object recognition rate, accurate pose estimation result. Moreover, it exhibits the capability of dealing with environmental illumination changes.

4

DENSE AND DEFORMABLE MOTION EXTRACTION OF DYNAMIC SCENE

In this chapter, we present a novel hierarchical MRFs optimization method for dense and deformable motion extraction in dynamic RGB-D scenes. In particular, this hierarchical MRFs structure consists of two layers, respectively named segmentation and correspondence layer. Firstly, in the segmentation layer, the dynamic foreground data is successfully segmented through a pixel-level MRF. Secondly, in the correspondence layer, the extracted foreground data is constructed as a 3D point-level MRF. A new surface descriptor named deformable color and shape histogram is proposed. It is combined with photometric and geometric features to represent a deformable surface. The foreground data correspondences across consecutive frames are extracted next. Finally, the dynamic scene motion is retrieved correctly from these correspondences. The discrete optimization scheme is utilized for binary classification and multi-labeling problems in these two layers. Moreover, a dataset of dynamic RGB-D scenes is built, which involves different motion patterns and surface properties of dynamic foreground. The effectiveness and efficiency of our proposed approach for highly accurate foreground segmentation and motion extraction is validated in experiments.

The remainder of this chapter is organized as follows: Section 4.1 to Section 4.5 provide detailed system framework, construction of hierarchical MRFs structure, design of DCSH descriptor and optimization schemes. The experimental results including dynamic RGB-D scene sequence dataset building, accuracy analysis and evaluation of runtime performance are presented in Section 4.6. Finally, Section 4.7 summarizes this work.

4.1 FRAMEWORK OF DYNAMIC SCENE MOTION EXTRACTION

We propose a system framework that estimates the dynamic foreground data and 3D motion fields of all foreground points by extracting their correspondences across consecutive RGB-D image frames. It is used to retrieve the dynamic scene motion and detailed surface deformations.

As shown in Figure 4.1, for each RGB-D sequence, background model needs to be learned in the initial phase. After that, the dynamic foreground data is extracted based on differences to the learned background scene model and previous image frame. We track all the extracted foreground points using correspondences labeling across con-

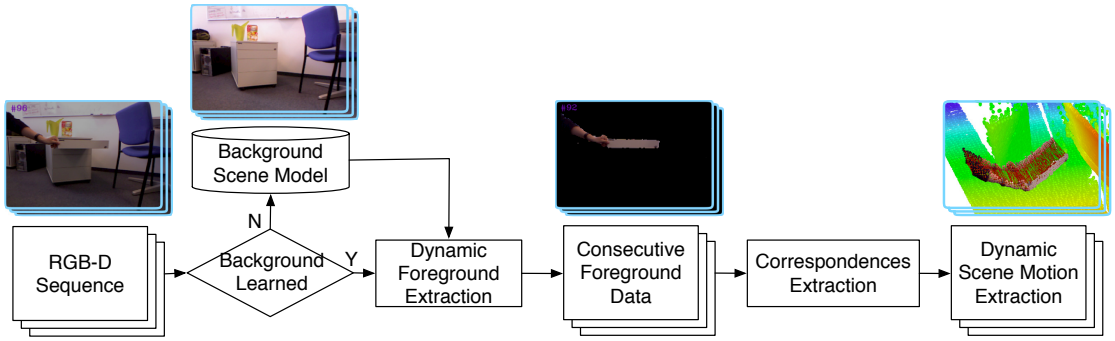


Figure 4.1: Flowchart of dynamic scene motion extraction system using hierarchical MRFs.

secutive frames under global optimization scheme. Finally, the consecutive dynamic scene motion is estimated from the 3D displacements of corresponding points. This way, we can follow the possible large displacement in the foreground data motion and can calculate the detailed surface deformation at the same time.

4.2 HIERARCHICAL MRFS STRUCTURE DESIGN

Aiming to extract dynamic foreground and retrieve motion from correspondences of foreground data across consecutive frames, we design a hierarchal MRFs optimization method for respective binary classification and multi-labeling problems.

4.2.1 *Markov Random Field Basics*

Markov Random Field (MRF) has been frequently employed for different computer vision problems. These problems can be modeled as a graphical model and optimally posed as Bayesian labeling using the maximum a posteriori (MAP) probability estimation. Let a graph be defined as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ denotes the graph vertices and $\mathcal{E}$ edges. $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is antisymmetric and antireflective. An ordered pair $(s, t) \in \mathcal{E}$ represents an edge connection. It is necessary to find an optimal configuration of the graph $\mathcal{G}$ that assigns the label set L to all nodes s. For different vision problems, different energy functions are minimized which corresponds to a maximum a posteriori configuration.

4.2.2 *Hierarchical MRFs Structure in Different Layers*

As shown in Figure 4.2, our proposed hierarchical MRFs consists two layers: segmentation and correspondence layer. In particular, segmentation layer is designed for the dynamic foreground extraction, and higher correspondence layer is designed to main-

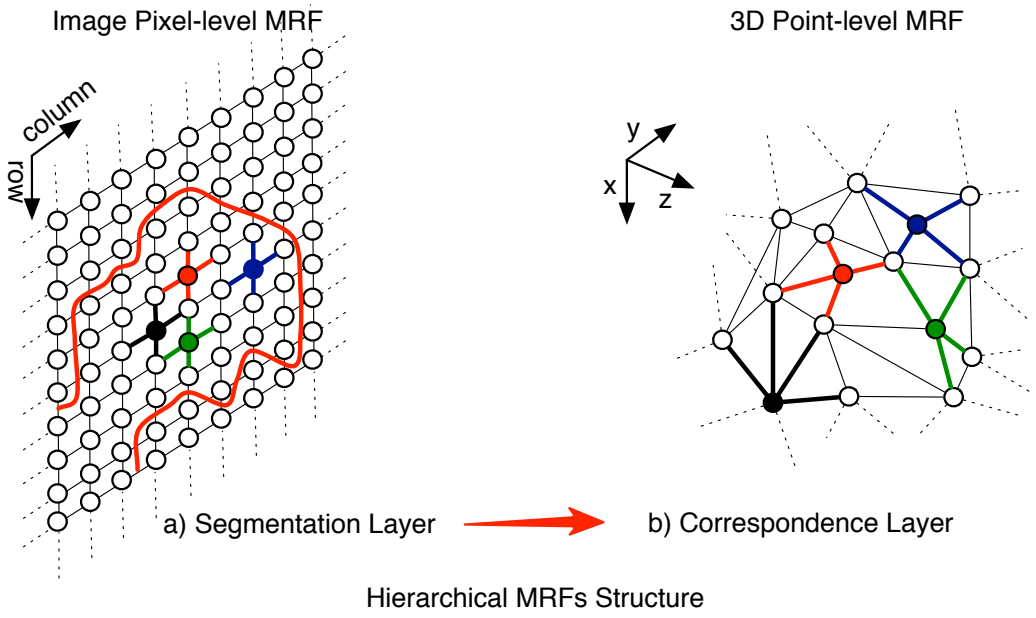


Figure 4.2: Hierarchical MRFs structure: In the segmentation layer, all pixels are well structured by 2D image coordinates (column, row). This image pixel-level MRF is built with the local neighbors in image's column and row directions; In the higher correspondence layer, all 3D RGB-D points of the extracted dynamic foreground from the segmentation layer, are not well structured. This 3D point-level MRF is built by searching the nearest neighbors in 3D Euclidean space.

tain the correspondences of extracted foreground data across consecutive frames. In the segmentation layer, all pixels are well structured by their 2D image coordinates. This image pixel-level MRF is built on the local image neighbors in column and row directions. The 3D points in the correspondence layer are not well structured. Hence, this 3D point-level MRF is needed to built by searching the nearest neighbors in 3D Euclidean space.

In segmentation layer, we formulate this dynamic foreground extraction problem as a binary classification problem. Each node representing a pixel in the image needs to be labeled as foreground or background in the label set $L = \{fg, bg\}$. In correspondence layer, we formulate the foreground correspondence problem as a multi-labeling problem. The extracted consecutive dynamic foreground pair is built as 3D point-level MRF pair $< F_t, F_{t-1} >$. These two corresponding MRFs may contain different number of foreground points. We define the MRF which has less nodes as the source $F_{source} = min_{num}(F_t, F_{t-1})$, and the one with more point as the target MRF as $F_{target} = max_{num}(F_t, F_{t-1})$. This ensures the source finds unique correspondence with the target. Therefore, the node set $S = \{s_0...s_m\}, m = num(F_{source})$ and the discrete label set $L = \{l_0...l_n\}, n = num(F_{target})$ are defined for the MRF pair corresponding process.

4.3 DYNAMIC FOREGROUND EXTRACTION

Firstly, the dynamic foreground data is necessary to be extracted from the RGB-D dynamic scene sequences. It is formalized to label each node of MRF as foreground or background as a binary segmentation problem.

As the input data, the synchronized RGB and depth images are given with frame oder in entire dynamic sequence. The MRF for segmentation layer is structured by the image coordinates. At each pixel position (P_x, P_y), the node in MRF has the information of (r, g, b, x, y, z). The edge of MRF nodes is defined as the connection with its four spatial nearest neighbors at pixel positions $(P_x \pm 1, P_y)$ and $(P_x, P_y \pm 1)$.

In this dynamic foreground segmentation problem, we find an optimal configure $\widehat{\Omega}$ that assigns a label l_i for each node s_i in $\mathcal{G}$, so that following energy function is minimized:

$$E(\Omega|\Theta) = \sum_{s_i \in \mathcal{V}} \Theta_s(l_i) + \lambda \sum_{s_i \in \mathcal{V}} \sum_{(s,t) \in \mathcal{E}} \Theta_{st}(l_s, l_t), \tag{4.1}$$

where Θ_s is the unary potential representing the data similarity that s is classified as foreground or background, Θ_{st} is the pairwise potential representing the smoothness penalty that s and t are signed as different label, and λ is a suitable weighting coefficient of smoothness penalty.

4.3.1 *Data Term from Image Similarity*

The data term Θ_s measures the similarity or likelihood that node s is classified as the possible label $l_i \in (fg, bg)$, and calculated as:

$$\Theta_s(l_i) = \begin{cases} 1 - D_s & \text{when } l_i \text{ is background} \\ D_s & \text{when } l_i \text{ is foreground} \end{cases} \tag{4.2}$$

where D_s is the node data similarity to the class "foreground".

The dynamic foreground data is defined as the points that have large differences between the past and current frame. These points represent the moving objects in scene and are our interests to estimate their 3D motion field. During the movements of foreground points, the occluded background scene model data in the current frame are also possible to be viewed as the dynamic foreground. Nevertheless, these static data is not of any interests for the final corresponding process. The final data term needs to involve not only the consecutive frame data similarity term D_F, but also the background scene model similarity term D_B. The average filter-based method is

utilized for background scene learning in our system framework [130]. These two terms D_F and D_B need to be combined to represent the final D_s:

$$D_s = min(D_F, D_B). \tag{4.3}$$

From the source point data, all the color (r, g, b) and depth (x, y, z) information is considered to compute the similarity.

$$\begin{aligned} D_F &= \alpha f^F_{color} + (1-\alpha) f^F_{depth}, \\ D_B &= \alpha f^B_{color} + (1-\alpha) f^B_{depth}, \end{aligned} \tag{4.4}$$

where α is a weight coefficient, f_{color} and f_{depth} are the functions to calculate color and depth differences. For the color differences, RGB information is converted into HSV space, and only H and S components are utilized to make the differences independent of the brightness [42]:

$$f_{color} = exp(- \parallel (S\cos(2\pi H), S\sin(2\pi H)) - (S'\cos(2\pi H'), S'\sin(2\pi H')) \parallel), \tag{4.5}$$

and for the depth distortion, the euclidian distance with two 3D points $p = (x, y, z)$ is used:

$$f_{depth} = exp(- \parallel p - p' \parallel^2 / \sigma^2), \tag{4.6}$$

where σ^2 is the variance of this 3D Euclidean distance, (HS,H'S') and (p, p') represent the relevant nodes at same pixel position in the consecutive frames for D_F calculation, and at same pixel position in current frame and the learned background scene model for D_B calculation.

4.3.2 *Smoothness Term Calculation*

The smoothness term Θ_{st} captures the spatial continuity between neighboring pixels. It gives certain penalty if these two pixels within neighborhood are assigned as different label. If s and t are the connected nodes in $\mathcal{G}$. Consequently, the smoothness term is defined as :

$$\Theta_{st}(l_s, l_t) = \begin{cases} \tau & \text{if } l_s \neq l_t \\ 0 & \text{if } l_s = l_t, \end{cases} \tag{4.7}$$

where τ is a constant penalty. As our definition, it equals to the largest data term value as $\tau = max_i(D_{s_i})$.

4.4 CORRESPONDENCES LABELING FOR FOREGROUND PAIR

The dynamic foreground has been extracted in the segmentation layer MRF (see Figure 4.2a) for each frame of RGB-D sequence. The 3D point-level MRF is built in the correspondence layer from our hierarchical MRFs, to find all the foreground correspondences across consecutive frames (Figure 4.2b). Currently, we focus on the correspondence problem between two adjacent frames. Each frame's foreground data is a set of colored 3D points $p = (r, g, b, x, y, z)$. Note that the raw foreground point clouds need to be converted into voxel grids at first. On one side, it downsamples the raw high density data. On the other, it deals with the image scaling problem for the building of MRF structure. In particular, more colored point clouds will be captured when an object is moving closer to the sensor. In this case, the points representing the moving object should have the same number. This scaling problem needs to be solved for many vision applications.

In correspondence layer, the MRF network is built by nearest four neighbors in 3D Euclidean space. The consecutive frames are classified as source and target MRF based on the number of their nodes respectively. In the source MRF, each node s has its own identity as a set $S = \{s_0 ... s_m\}$. We define the target MRF's node identity space $\{0...n\}$ as corresponding label space $L = \{l_0 ... l_n\}$, where $n \geqslant m$. After the correspondence process, each node in the source MRF gets its relevant corresponding identity of node in the target MRF. Consequently, the displacement of the point across consecutive frames can be extracted and converted into 3D motion field.

This multi-labeling problem for correspondences extraction is formalized to find an optimal configure $\widehat{\Pi}$ that assigns a certain label l_i for each source node s_i in the source MRF. The label l_i is in the identity space L of node set T in the target MRF. A global energy function needs to be minimized as following:

$$E(\Pi|\Phi) = \sum_{s_i \in \mathcal{V}} \Phi_s(l_i) + \beta \sum_{s_i \in \mathcal{V}} \sum_{(s,t) \in \mathcal{E}} \Phi_{nb}(l_s, l_t) + \gamma \sum_{s_i \in \mathcal{V}} \sum_{s_j \in \mathcal{V}} \Phi_{ocp}(l_i, l_j), \quad (4.8)$$

where Φ_s is the unary potential representing the data similarity between source node s_i and target node t_{l_i}, when s is corresponded to an id label l_i. Φ_{nb} is the pairwise potential representing the neighborhood constrains. The corresponded nodes should also be neighbors in the target MRF when labels of s and its neighbor t are assigned. Φ_{ocp} is the pairwise potential representing the correspondence occupancy constrains, which safeguards that each source node s should not be assigned as the same identity label. β and γ are suitable weighting coefficients for the respective potentials.

In order to present the MRF node data comprehensively and efficiently, we design a new surface descriptor named *Deformable Color-Shape Histogram* (DCSH) which com-

bined its shape and color features. This new descriptor is an extension of the work presented in [161, 162].

4.4.1 *Deformable Color-shape Histogram*

The deformable color-shape histogram is proposed to describe each node, based on its surrounding surface information. This is necessary to utilize comprehensive photometric and geometric components. This novel descriptor is combined with shape and color features from the raw colored point clouds, whose center is the node.

4.4.1.1 *Shape Features*

The entire correspondence process needs to have the capability to deal with deformable foreground data. The surface deformation of foreground might be caused by inlier articulated or bending motion. As shown in Figure 4.3, the foreground data of a paper is extracted when it is moving and its surface is bending in the meanwhile. Geodesic distance is employed here for the deformable surface's shape feature of node s. In particular, geodesic distance captures the geometry of point clouds and automatically adapts to deformation [88]. At first, the entire foreground point clouds is built as an adjacency map using euclidean distance-based k-nearest neighbor searching. The node s of our MRF network represents a vertex of point clouds. Dijkstra's algorithm is used for searching its shortest paths to target point and extracting their pairwise geodesic distances $D_{geo} = \{g_i\}$.

Dijkstra's search algorithm is one of the most popular algorithm for single source, shortest path estimation. A node-connected graph as a adjacency-map is required at first. With the index of the source and target nodes, it operates as an iterative algorithm to get the shortest path between them.

The entire process can be divided into three steps: 1) preprocessing; 2) distance computation; 3) reasoning. The preprocessing part is visited only once in the beginning of the execution. All the nodes of the graph except the source node are marked as unvisited, and the distance to them is set to infinity. The distance of the source node is set to zero, and the node itself is tagged as current. In the computation part, the distance between the current node and all of its adjacent unvisited nodes is computed. Furthermore, if the newly computed distance to the adjacent node is shorter than the one already present, it will be updated with the new value. The current node is tagged as visited and the node with the smallest path distance is tagged as current. This is followed by the reasoning part where the current node is compared to the target node. If the match is reached then the system returns, if not the computation part for the new current node is executed then. The detailed algorithm is presented in Algorithm 4.1.

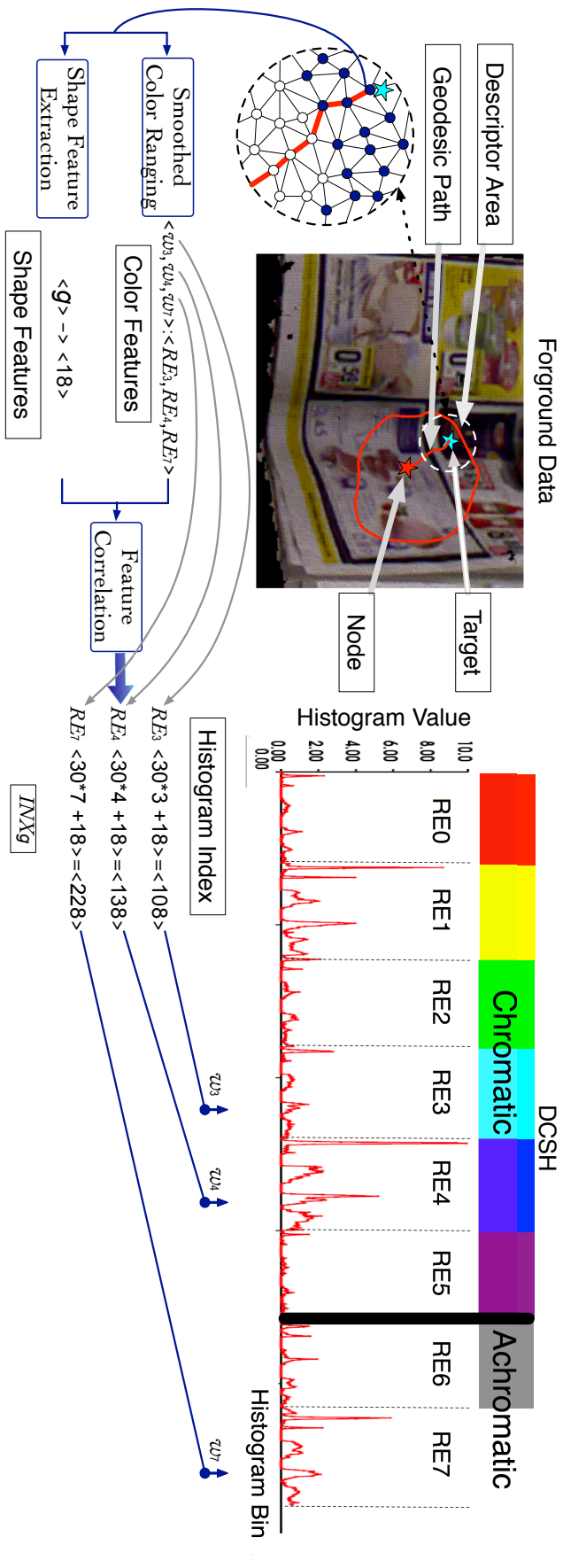


Figure 4.3: Generation of a DCSH using color and shape features of surrounding surface, whose center is an MRF node (red star). Each node data is represented as a DCSH.

Algorithm 4.1 Dijkstra's search algorithm for finding the shortest path in a adjacency map (graph). The algorithm takes as an input a connected graph, the index of the source node and the index of the target node.

```
# build an adjacency map using k-nearest neighbor searching
G ← adjacency map;
# get the source and target node indexes
source ← source node index;
target ← target node index;

# initialization of distance and visited label vector for each node
for all V ∈ G do
   dist[V.index] ← inf;          # initialize as infinite
   visited[V.index] ← false;     # initialize as not visited
end for
dist[source] ← 0;
C ← GraphPath[start];            # graphical path as a vector of index
P_C ← distance(C);               # total distance from visited path

# if reach the target node, stop iterating and return the geodesic distance
if C.index = target then
   return dist[C.index];
end if

# search the shortest path based on current node's neighbors
for all N ∈ V.neighbors do
   # guarantee it has not been visited
   if not visited[N.index] then
      N_D;                       # distance with current neighbor
      d ← P_C + N_D;             # update the distance
      if d < dist[N.index] then
         dist[N.index] ← d;
      end if
   end if
end for

visited[C.index] ← true;         # assign as visited
# update the shortest path
C ← GraphPath[index(min(dist[]))];

# continue computing until it find the target node
goto 7;
```

Given a geodesic distance threshold, a colored point set (represented as deformable surface) around s is extracted to build its final DCSH descriptor. Illustrated by Figure 4.3, the red star represents the node s, the cyan star is one target point in s's descriptor area (red circle) and the red line is its geodesic path from s.

4.4.1.2 *Color Features*

The HSV value of each point is used to estimate color contributions both in chromatic and achromatic area. The chromatic area is considered as the true color space, and achromatic area represents the gray scale space in the whole SV space. Eight histogram regions RE with index $u = \{0 \cdots 7\}$ are defined to build the entire DCSH. Six of them are used for chromatic area, and the other two are for achromatic area. According to the HSV value, each point p is ranged into two adjacent histogram regions $\langle RE_u, RE_{u+1} \rangle$ in chromatic area, and one region RE_6 or RE_7 in achromatic area, with respective contributions being $\langle w_u, w_{u+1}, w_6 | w_7 \rangle$. For more details of color ranging process, see Section 3.3.1.

4.4.1.3 *Shape and Color Feature Correlation*

An example of generating DCSH using color and shape features is illustrated in Figure 4.3. Constant histogram bin range for each shape feature g in each RE is given firstly. Then g is indexed into a final DCSH bin as INX_g. The color contributions $\langle w_u, w_{u+1}, w_6 | w_7 \rangle$ of each point are added at its shape feature's indexes in the corresponding three histogram regions $\langle RE_u, RE_{u+1}, RE_6 | RE_7 \rangle$. Each node's shape feature g has 30 bins IN_g for each histogram region, and final bin index in DCSH regarding to each RE_h:

$$\begin{aligned} IN_g &= \lfloor \frac{30(g - g_{min})}{g_{max} - g_{min}} \rfloor, \\ INX_g &= IN_g + 30h, \end{aligned} \tag{4.9}$$

where $h \in \{u, u+1, 6, 7\}$ as the range id. Finally, these extracted color and shape features are correlated as a DCSH to represent the node s's descriptor.

4.4.2 *Data Term from DCSH Similarity*

The data term Φ_s represents the data similarity between the source node s and the target node t_l, where l is s's corresponding label which associated with the relevant node id in the target MRF. Bhattacharyya distance of node s's DCSH descriptor DH_s

and its corresponded node t_l's DCSH descriptor DH_{t_l}, is used to represent the data term:

$$\Phi_s(l_i) = \sqrt{1 - \frac{1}{\sqrt{\sum_{hi} DH_s \sum_{hi} DH_{t_l}}} \sum_{hi} \sqrt{DH_s(hi) \cdot DH_{t_l}(hi)}}, \tag{4.10}$$

where hi is the histogram bin of each DCSH.

4.4.3 *Neighborhood Constrain Term Calculation*

The neighborhood constrain term Φ_{nb} is proposed to guarantee that, the source node s and its neighbor t to be as close as possible in the target MRF network T for corresponding. We use Euclidean distance $\Delta(p1, p2) = \| p1 - p2 \|$ to evaluate the neighborhood likelihood. Therefore, this neighborhood constrain term as a pairwise interaction potential function is defined as:

$$\Phi_{nb}(l_s, l_t) = \begin{cases} \frac{|\Delta(T_{ls}, T_{lt}) - \Delta(s,t)| - min_{i,j}\Delta(T_i, T_j)}{max_{i,j}\Delta(T_i, T_j) - min_{i,j}\Delta(T_i, T_j)} & \text{if } l_s \neq l_t \\ \tau_{nb} & \text{if } l_s = l_t, \end{cases} \tag{4.11}$$

where τ_{nb} is a constant to prevent two nodes s and t, $(s,t) \in \mathcal{E}$, when these edge-connected nodes are assigned as the same correspondence label. Here, we set $\tau_{nb} = max_i \Phi_{s_i}$ which is the maximum value of the data dissimilarity.

4.4.4 *Occupancy Constrain Term Calculation*

The occupancy constrain term Φ_{ocp}, also a pairwise potential, is utilized to prevent the two nodes s_i and s_j not to be assigned as the same label. It is similar with the τ_{nb} meaning in Equation 4.11. Instead, these two nodes are edge-connected, that are not the neighbors in MRF network as $(s_i, s_j) \notin \mathcal{E}$. Therefore, we define the occupancy constrain term as following:

$$\Phi_{ocp}(l_i, l_j) = \begin{cases} \tau_{ocp} & \text{if } l_i = l_j \\ 0 & \text{if } l_i \neq l_j, \end{cases} \tag{4.12}$$

where τ_{ocp} is also a constant and its value is same as τ_{nb}.

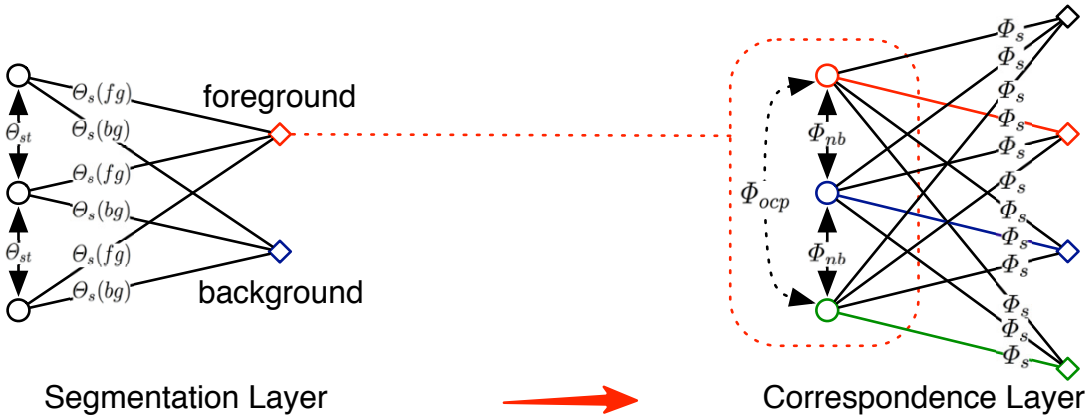


Figure 4.4: Global optimization scheme structure for energy minimization in different layers of the hierarchical MRFs.

4.5 OPTIMIZATION SCHEME FOR ENERGY MINIMIZATION AT DIFFERENT MRF LAYERS

After the hierarchical MRFs structure is built, an optimization scheme is necessary to minimize the energy functions in the segmentation layer and also in the correspondence layer as in Equation 4.1 and Equation 4.8 respectively. We adapt the *sequential tree-reweighted message passing* (TRW-S) algorithm [80], that has been developed for discrete energy minimization recently. In particular, TRW-S adjusts the message updating schedule and yields a lower bound guaranteed no to decrease. The energy is solved iteratively until convergence or a maximum number of allowed iterations is exceeded. Thus it is preferable compared to the *belief propagation* (BP) algorithm in which scheduling is heuristic and convergence is not guaranteed [143].

The optimal scheme is shown in Figure 4.4. In the segmentation layer, each node will be given data term Θ_s for different classes (foreground, background) and the smoothness term Θ_{st} for its neighbors when they have edge connection $\mathcal{E}$. The data term module is defined as $Seg :: NodeData(\Theta_s(fg), \Theta_s(bg))$. The binary pairwise data $\langle l_s, l_t \rangle$ is converted into $\langle (bg, bg), (bg, fg), (fg, bg), (fg, fg) \rangle$ as a four-dimension vector. Thus the smoothness term module is defined as $Seg :: EdgeData(0, \tau, \tau, 0)$ from Equation 4.7.

In correspondence layer, each node will be given the data term Φ_s, neighborhood contain term Φ_{nb} and occupancy contain term Φ_{ocp}. We define the data term module as $Corres :: NodeData([\Phi_s(l_0), ..., \Phi_s(l_n)])$, where n is number of label. The node s and its neighbor node t contributes the neighborhood constrain term. This pairwise data is converted into a vector V_{nb} with the size as $n \times n$, where $V_{nb}[l_t + l_s \times n] = \Phi_{nb}(l_s, l_t)$. Thus the neighborhood constrain module is defined as $Corres :: EdgeData(V_{nb})$, if two nodes s_i and s_j have no edge connection. We need to define the occupancy constrain module as $Corres :: OcpData(\tau_{ocp})$ from Equation 4.12.

4.6 EXPERIMENTAL RESULTS

To demonstrate the effectiveness of the proposed method, we perform a large variety of tests on RGB-D sequences involving different motion patterns and object surface properties of the dynamic foreground. Static Kinect sensor is used to record each sequence involves calibrated and synchronized color and depth images in time series within 30Hz. The sequence is reconstructed as ROS BAG[1].

We collect a dataset of various dynamic RGB-D scenes consisting of different background, various dynamic motion patterns and foreground data with different surface properties. The entire dataset will be released in the near future. In this work, three sequences with different features are chosen to present the experimental results and evaluations (see Figure 4.5, Figure 4.6 and Figure 4.7): The "DRAWER" sequence involves the motion of manipulating a drawer, that involves a prismatic articulation and its foreground data is relative rigid; The "PAPER" sequence collects the motion of waving a paper, that its motion is complex; The "CLOTH" sequence collects the motion of moving a hand-hold cloth, whose foreground data is strongly deformable.

Original RGB and depth images' resolution are both 640×480. In our experiments, we synchronize and downsample these images into half. Firstly, pixel-level MRF is built for dynamic foreground data extraction. After that, we convert these image pixel-level foreground data into colored point clouds. And then, the 3D point-level MRF is built for foreground correspondences across consecutive frames, finally extract the dynamic motion. Note that our implementation is partially based on Point Cloud Library[2]. The sequences of our dataset contain enough initial frames at the beginning for background model building. The dynamic foreground data is extracted from image points with constant number as $320 \times 240 = 76800$. In these entire sequences, the dynamic foreground data contains 200 to 600 colored voxels for correspondence in general. For discrete optimization in segmentation and correspondence layer in our hierarchical MRFs structure, we set the iteration times to 15 and 20 respectively with the consideration of accuracy and computational cost.

We present the experimental results through these three sequences, to emphasize the benefits and uniqueness of proposed approach, that can deal with complex motion, dynamic rigid and deformable foreground data. As shown in Figure 4.5, Figure 4.6 and Figure 4.7. the dynamic foreground data is correctly segmented from the background, and foreground pairs across consecutive frames (four frames distance are used here) are corresponded afterwards. The red array represents the correct nodes correspondences and motion direction. In the "DRAWER" sequence (Figure 4.5), the hand's and drawer's motion are obviously different. It is represented as different correspondences

1 http://wiki.ros.org/rosbag
2 http://www.pointclouds.org

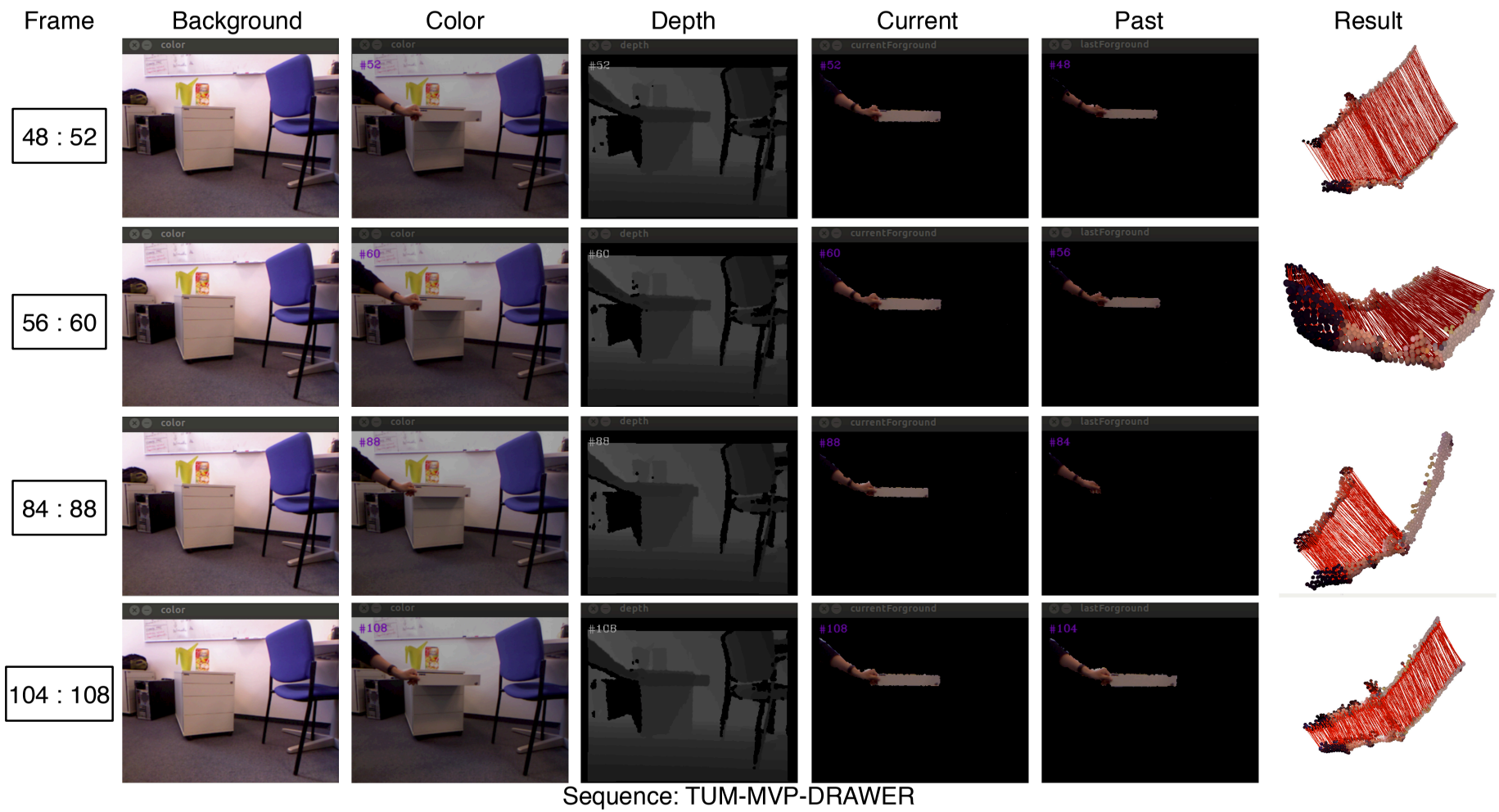


Figure 4.5: Experimental results on different testing sequence: TUM-MVP-DRAWER.

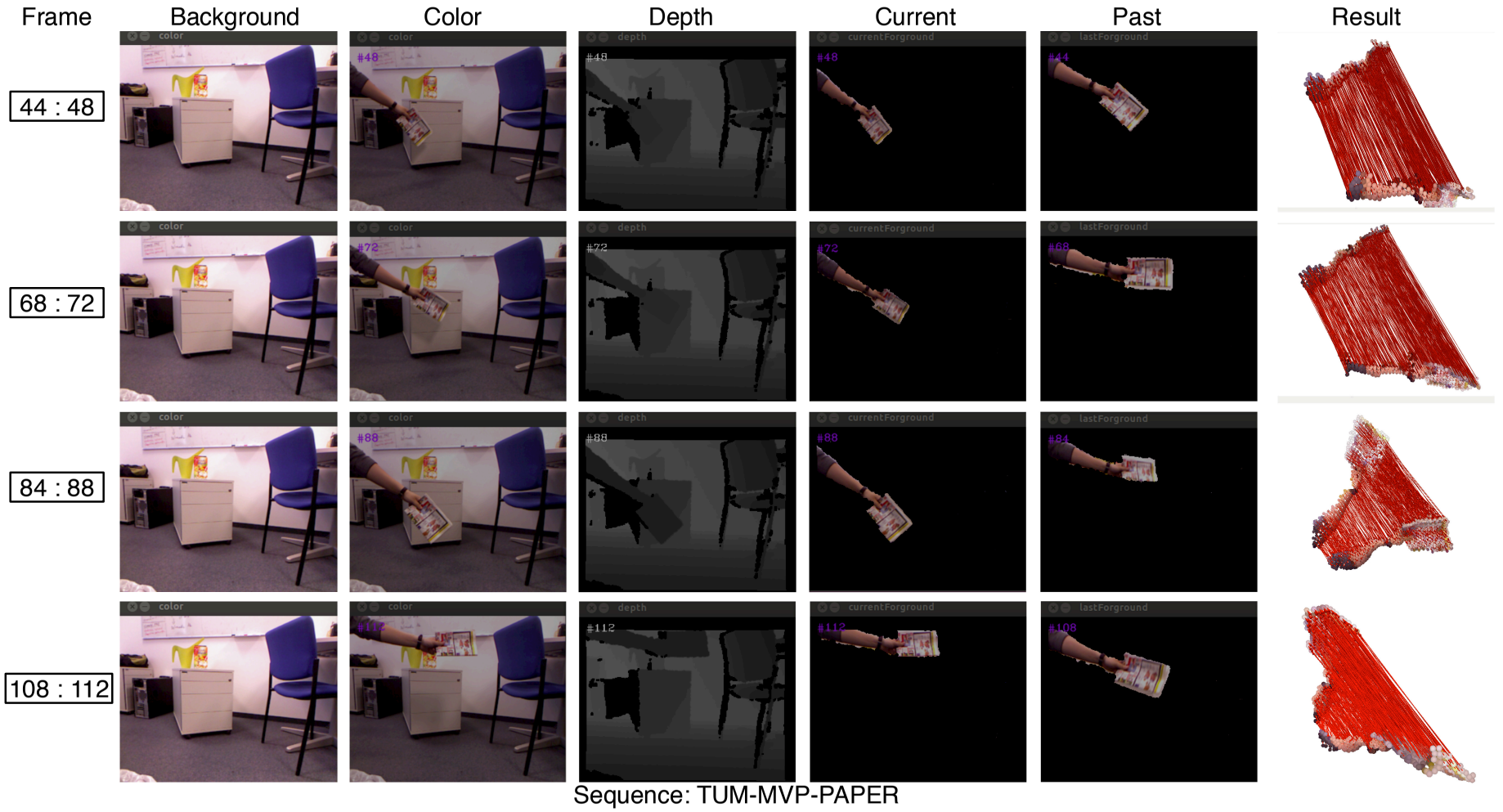


Figure 4.6: Experimental results on different testing sequence: TUM-MVP-PAPER.

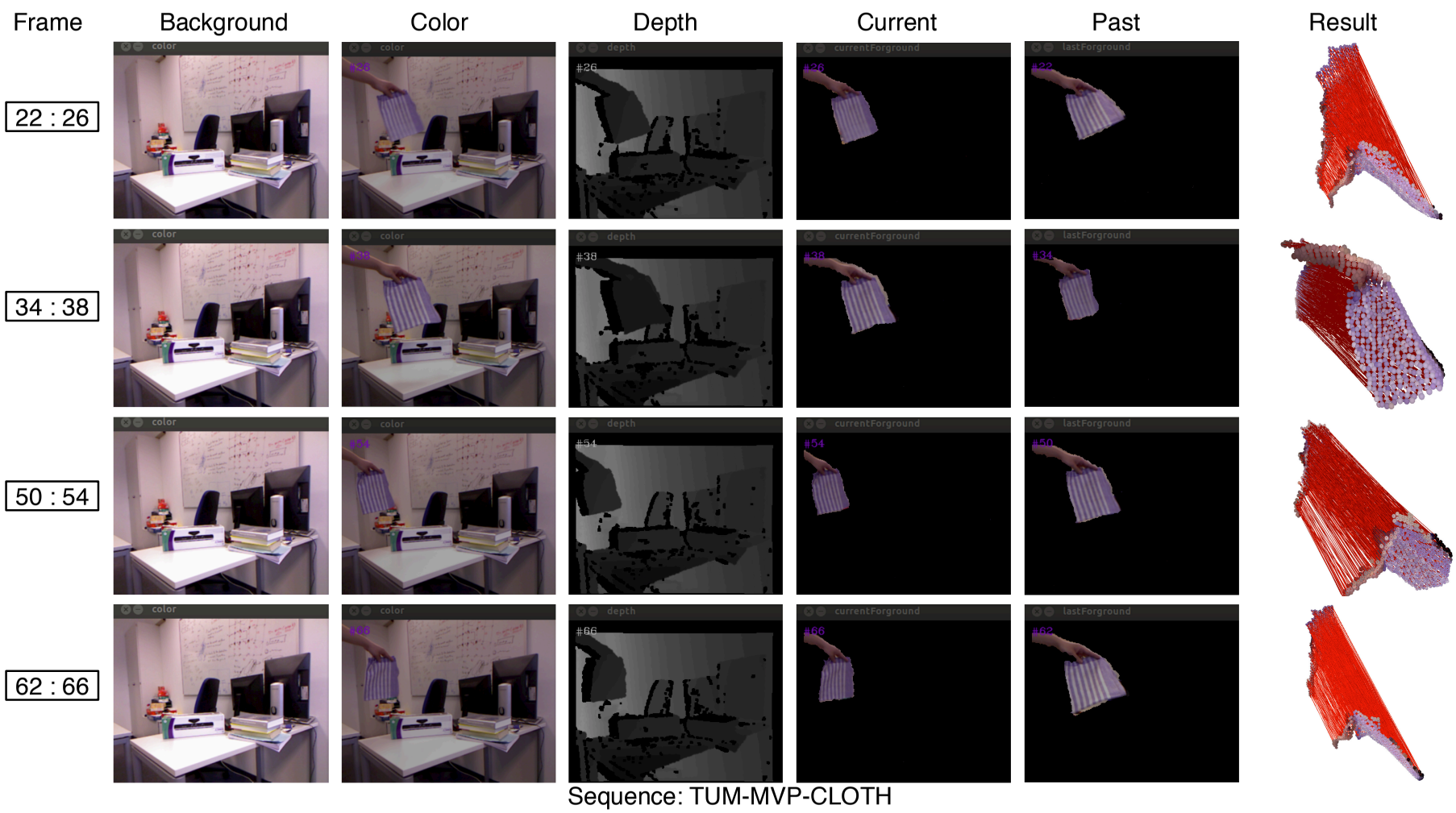
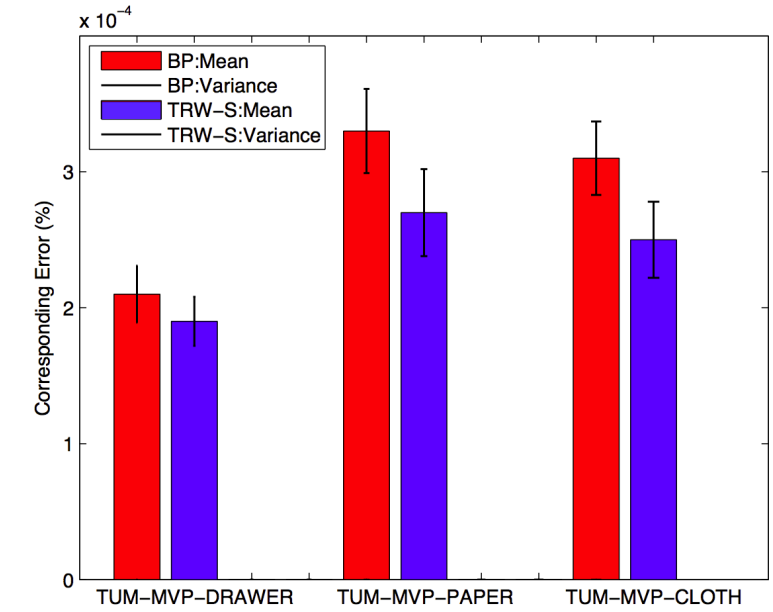
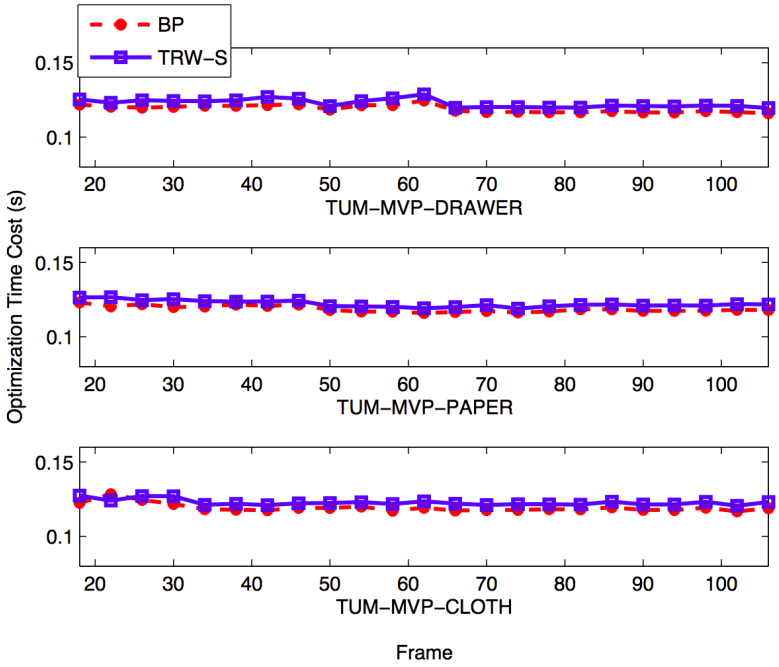


Figure 4.7: Experimental results on different testing sequence: TUM-MVP-CLOTH.



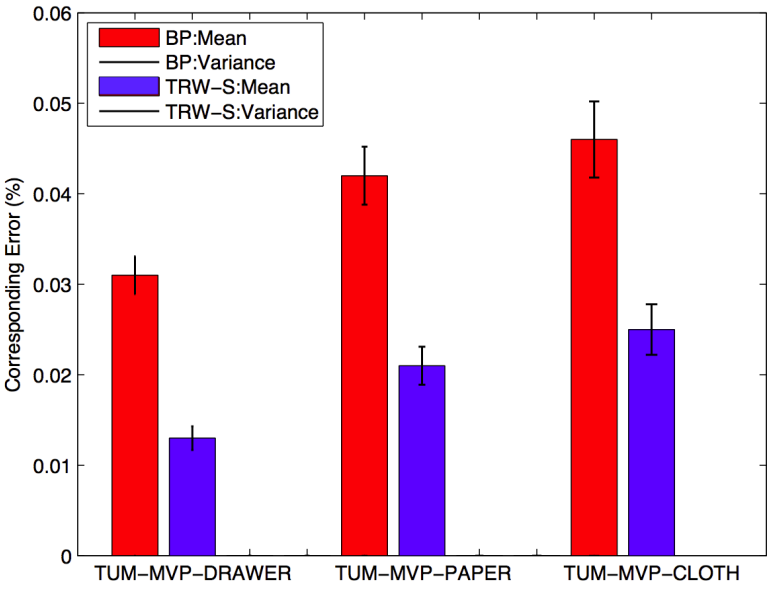
a) Segmenting Error



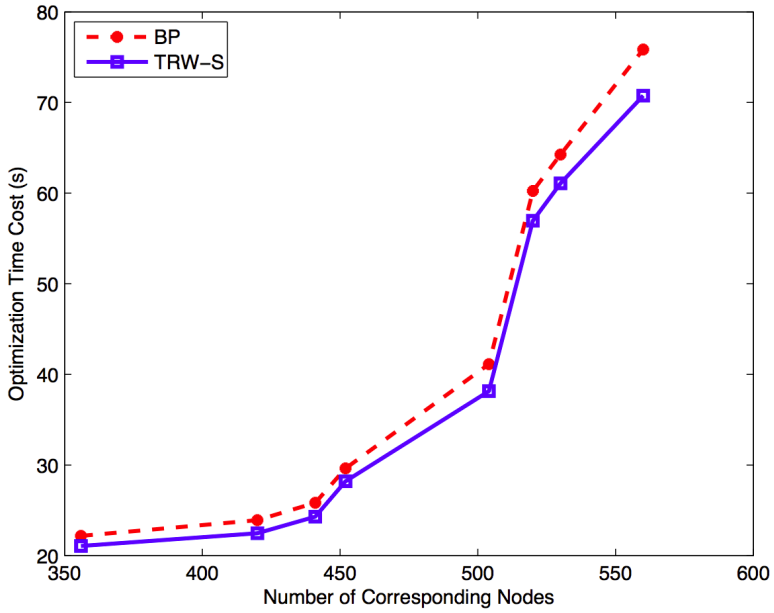
b) Segmenting Runtime Performance

Figure 4.8: a) Segmentation error of different testing sequences. b) Runtime performances for segmentation of different sequences.

distances across consecutive frames. The dynamic motion field is extracted and can be used for the rigid segmentation, motion pattern classification and articulated object modeling. At frame pair $< 84 :: 88 >$, our system still can retrieve correct results even the dynamic foreground data dramatically changed. In the "PAPER" sequence (Figure 4.6), the correct correspondences are extracted, when moving up and down



a) Corresponding Error



b) Corresponding Runtime Performance

Figure 4.9: a) Correspondence error of different testing sequences. b) Runtime performances by different nodes numbers for corresponding.

at frame pair $< 84 : 88 >$ and with large displacements at frame pair $< 108 : 112 >$, that cannot be solved by optical flow and other motion model based methods. In the "CLOTH" sequence (Figure 4.7), the foreground data deforms strongly during the motion. The correct correspondences results present our proposed method's efficiency and stability. Especially, our deformable color and shape histogram (DCSH) can describe this kind of deformable objects correctly.

To better evaluate our proposed approach, the ground-truth data is first manually labeled and quantitative evaluations are provided. We analyze the error mean and variance of incorrect corresponded node percentage for different sequences. Belief Propagation (BP) is also utilized for comparison within the same optimization data structure as TRW-S. For evaluation of segmentation result evaluation, the error involves the false positives and false negatives percentage of entire image points. Figure 4.8a shows the mean error and error variance of different test sequences and Figure 4.8b illustrates their time costs. These two optimization methods work almost with same quality. Under 0.0004% of 76800 image points are segmented falsely. Because of the constant node number and binary label set, the time cost is preserved under 0.13 ms for all sequences. For correspondence result evaluation, as shown in Figure 4.9a, TRW-S outperforms BP obviously. In general, the performance of our system is presented as mean error under 0.03% and error variance under 0.01%. For the runtime performance analysis, the time cost for different number of corresponding nodes are presented in Figure 4.9b. Since the cost is influenced by nodes number and label set size, runtime cost is dynamic during entire sequence processing. For runtime performance evaluation and comparison, we select the label size as 50 more than nodes number in general, to analyze the performance using BP and TRW-S optimization methods. In general cases, the runtime cost is under 40 s to get the correct correspondences extraction of foreground pair. All experiments run on 4-core 2.8 Ghz i7 CPU and 8 GB of RAM.

From above experimental results, the hierarchical MRFs structure is proven its efficiency and stability by guaranteeing highly correct motion extraction. Because of our proposed deformable surface descriptor DCSH, the system can deal with the dynamic scenes that contain various motion patterns and surface properties of dynamic foreground.

4.7 SUMMARY

In this chapter, we presented a novel hierarchical MRFs optimization method for dense and deformable motion extraction in dynamic RGB-D scenes. This hierarchical MRFs structure consists of segmentation and correspondence layer, constructed with respective image pixel-level and 3D point-level MRF. The discrete optimization scheme is utilized under novel energy functions in two different layers. At first, the dynamic foreground data is segmented from entire image at each frame, and afterwards, the dynamic motion is retrieved correctly by corresponding these extracted foreground pair across consecutive frames. A surface descriptor DCSH is also newly designed to represent deformable surface of foreground data, combined with photometric and geometric features. Moreover, a dataset of various dynamic RGB-D scenes is built ef-

ficiently and will be released to public in the near future. Our proposed approach is proven to be effective and efficient by guaranteeing high accurate foreground segmentation and motion extraction from experimental results. Our method can easily be implemented in various higher level computer vision applications, such as 3D motion based object segmentation, articulated object modeling and deformable object analysis and so on.

5

ARTICULATED OBJECT MODELING BASED ON VISUAL AND MANIPULATION OBSERVATIONS

This chapter proposes an approach to model articulated object by integrating visual and manipulation information. Line-shaped skeletonization based on depth image data is realized to extract the skeleton of an articulated object, which are in different shape configurations. Using observations of the extracted object's skeleton topology, the kinematic joints of the object are characterized and localized. Robot end effector's force data in the form of task-space force are required to manipulate the articulated object. This data is collected by kinesthetic teaching and learned by Gaussian Mixture Regression in kinematic joint state space of object. Following modeling, manipulation skills for articulated object are realized by first identifying the current object joint states from visual observations and second generalizing learnt force to accomplish the new task. We validate the proposed method from the experimental results in different scenarios of an autonomous robot.

This chapter is organized as follows. From Section 5.1 to Section 5.5, We define the problem and propose a method to model an articulated object, mapping the learnt manipulation force into object joint space, and generation of manipulation skills based on recognized articulated object's joint state and goal of new task. The experimental setup and results are presented In Section 5.6. Finally, Section 5.7 summarizes this work.

5.1 ARTICULATED OBJECT MODELING BASED ON VISUAL AND MANIPULATION DATA

5.1.1 *Definition of Articulated Object Model*

Articulated object is defined as a combination of multi rigid parts which are connected with multiple kinematic chains. To manipulate articulated objects, comprehensive visual and manipulation information is required. This data is necessary to represent the structure, kinematic relationships and dynamic properties of unique articulated object . An articulated object can be described by its structure with number and type of kinematic joints, link properties and kinematic relationships between neighboring links of its rigid parts. Figure 2.5 shows some examples of articulated object. Basic geometry features which are used for rigid object modeling and recognition, such as Viewpoint

Feature Histogram (VFH) [119], are not suitable for deformable objects. However, these approaches require complete depth information of the object. Since articulated objects can lie in a practically huge number of different configurations, capturing information about all these potential configurations is practically infeasible. For this reason, object skeletonization is the most suitable method to extract the structure and kinematic joint constraints of an object. Moreover, for fast manipulation on articulated object, autonomous robots need to obtain the capability to generate the proper manipulation skills based on articulated object's joint state and can adapt with the different goal of new task. Consequently, to the best of our knowledge, we are the first to model an articulated object combining visual and manipulation observations. The detailed contents of articulated model is described as following:

$$Obj = (S,\ J_m(T,\ P,\ C),\ \mathbf{f}), m = 1, .., M \tag{5.1}$$

where S represents the skeleton of the object which is used for object recognition, J_m joint descriptor of the m-th joint, T joint type, P joint position and C joint constraints. The $\mathbf{f}(J_1, ..., J_M)$ is the Cartesian force which is needed to manipulate the object where $J_1, ..., J_M$ are joint descriptors of the articulated object where M is the number of joints.

5.1.2 *Manipulation Skills Formalization*

Investigating multiple-joint objects is highly complicated and implies sufficient modeling of all individual joints of the object. Because of this, in this work, we focus on modeling of single-joint articulated objects where visual and manipulation information is integrated for highly efficient object manipulation. The framework presented here can be extended to modeling multiple-joint objects though and this is going to be developed in near future. Manipulation force constitutes part of an object's model since it indicates the dynamic properties of the object. This force is critical to the success of a robotic task and depends on the object's current joint states. The manipulating force can be represented by $\mathbf{f} = \pi(s_{J_m},\ e),\ m = 1, ..., M$, where π is a force generation policy, s_{J_m} the state of the m-th joint which may describe the angle of a rotational joint or length of a prismatic joint, and e indicates the new task goal as the target state of relevant kinematic joint.

5.2 FRAMEWORK FOR ARTICULATED OBJECT MODELING

Figure 5.1 shows the framework which is used to model a single-joint articulated object. The framework consists of two main components. A database of articulated object models is built at first based on modeling method. For the real scenarios, the incom-

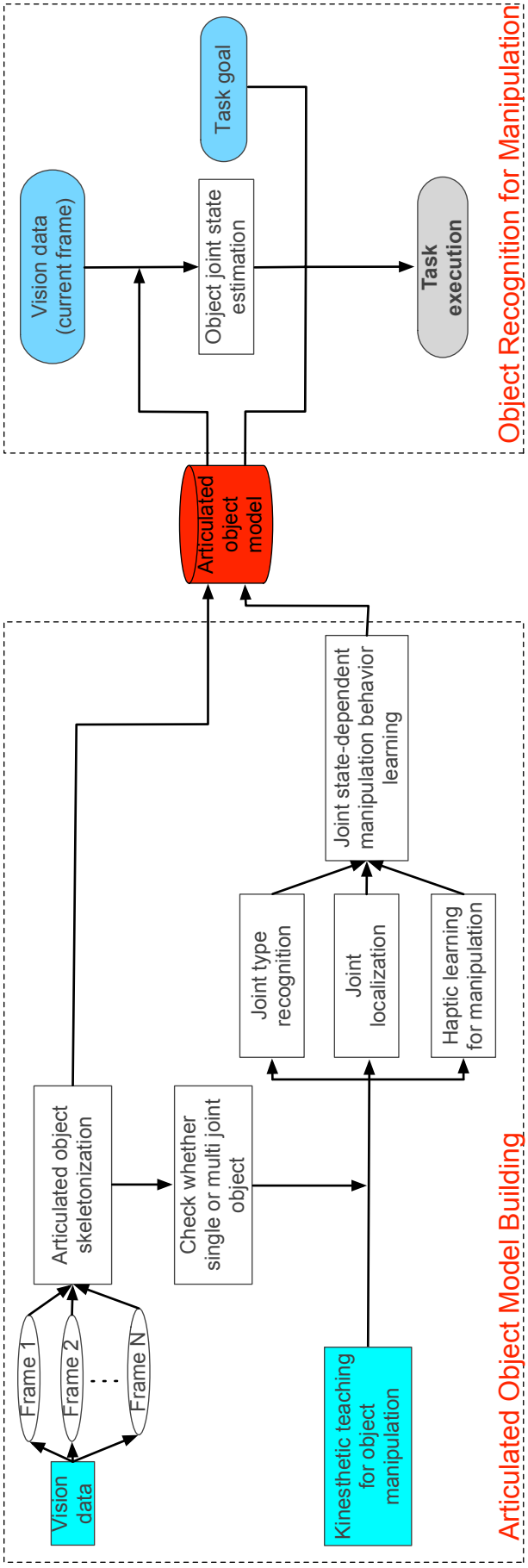


Figure 5.1: Proposed framework.

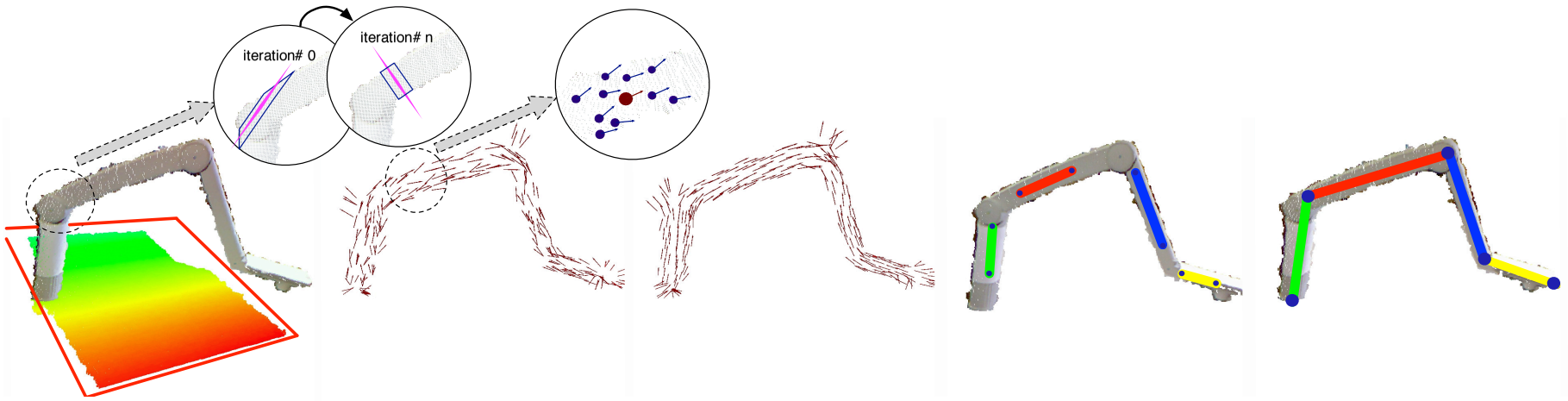
ing object can be recognized based on visual information and proper manipulation can be generated based on object joint state and goal of new task. The modeling stage can be divided into two parts where the first part involves vision-based object skeleton topology extraction and the second part consists of identification of the object's dynamic properties by teaching the robot appropriate force to operate the object. The kinematic joint properties (T, P, C) of a joint J are estimated from observation of the skeleton S across multiple configurations. Using learning by demonstrations, the appropriate force f is learned and mapped into the joint space of articulated object. The autonomous robot observes the articulated object and extracts its current joint state. Then the proper manipulation skills are generated based on the task goal such as the position or joint angle the object can finally reach and its current joint state.

5.3 OBJECT SKELETONIZATION FROM VISUAL OBSERVATION

A point cloud, in terms of depth image data representing the arc shape of object, is used for skeletonization for articulated object. This is realized by observing multiple frames of the object's kinematic links. The skeleton of the object is extracted and provides the capability for robot to automatically recognize the object and estimate its current joint states. Based on extracted object skeleton and the location of skeleton nodes, the object is classified as a single or multi-joint object. Skeleton models which represent the medial axis of a 3D model are widely used for object reconstruction and arterial object analysis. In [144], rotational symmetry axis is used for the object skeleton points estimation. That work requires the full range point cloud of the object and uses the assumption that all object's model is pipe-like. Instead, a novel method of skeletonization of articulated objects is presented in this work. Our proposed skeletonization method is not based on the assumption of pipe-like configuration. It can identify the objects from abstract structures even within a plane-like shape. As two examples of articulated object with different type of structure, the phone arm is pipe-like as shown in Figure 5.2 and the car's door is plane-like as shown in Figure 5.4.

5.3.1 *Vector Field Generation*

To simplify the problem, the articulated objects are assumed to be attached with a planar background. Firstly, the Random Sample Consensus (RANSAC)-based plane fitting algorithm is used to extract the point cloud of an articulated object [119], shown in Figure 5.2a and Figure 5.4b. After that, the vector field presents the best local rotational symmetry of each point in the extracted object point cloud. Our method extracts the vector field utilizing the optimized cutting plane. Based on RANSAC plane estima-



a) Object extraction based on RANSAC plane estimation

b) Vector field generation based on optimized cutting plane

c) Vector field smoothing based on Gaussian-weighted nearest neighbors

d) Line shape skeleton extraction based on vector field clustering

e) Skeleton topology estimation based on the line growth

Figure 5.2: Skeletonization steps of a multi-joint articulated object (phone arm).

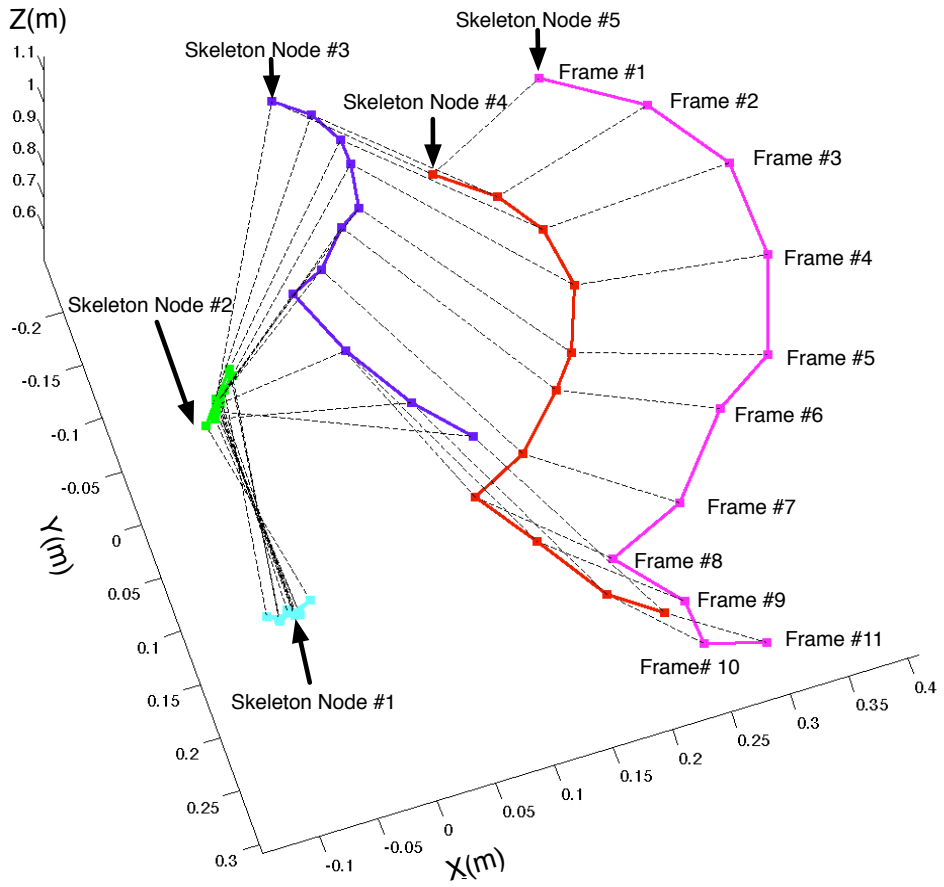


Figure 5.3: Skeleton node traces through different visual frames: black lines present the skeleton topology; each skeleton node trace is shown by a different-color solid line.

tion with a certain number of iteration steps T_c, the vector field over the entire point cloud is generated. The best cutting plane $C_c = plane[x_i, v_i]$ which goes through the point x_i with the normal $\hat{v}$ is estimated by minimizing the number of inliers which are within the distance d_c. In addition, these points are necessary to be in the same cluster N_i of the related point x_i using the geometric nearest neighbors searching. The process is described as following:

$$\hat{v_i} = \underset{v \in \Re^3, \|v\|=1}{\arg\min} \; num(\{j_{N_i} | \; \|c_j - C_c^{(t)}\| \leqslant d_c; x_j \in X_{raw}\}), \tag{5.2}$$

where $t \in [1, T_c]$ is the iteration index. Figure 5.2(b) shows the result where the circles show the iteration step. Note that the direction of optimized cutting plane can be inverse. However, this case does not influence the following process and the generation of vector field. These estimated directions are reorganized based on the coefficients of base plane as the background.

A Gaussian-weighted method is developed to smooth the estimated vector field. The point x_i with normal v_i has the neighbor cluster X_i with points number n, which

is determined by the distance threshold d_s. The weight function w is defined based on the gaussian contribution. This contribution is calculated by each neighbor's 3D distance respect to the point x_i:

$$\begin{aligned} w_j &= \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{1}{2\sigma^2}\|x_j - x_i\|^2), \\ v_{i:new} &= \frac{\sum_{j=1}^{n} w_j v_j}{\sum_{j=1}^{n} w_j}, \; x_j \in X_i. \end{aligned} \tag{5.3}$$

In particular, the standard deviation $\sigma = 1$ is used in this work. Figure 5.2c and Figure 5.4c show the smoothed vector fields over the point cloud of object within different shape.

5.3.2 *Line-shape Skeleton Estimation*

The skeleton of the object is represented as the combinations of straight lines and linked nodes. These nodes are named as skeleton nodes. After smoothing, the vector field is clustered using the nearest neighbor clustering method [160]. We consider the positions and the directions of the estimated vector field for object skeleton extraction. At the same time, the object's skeletal point position can be estimated using the center of the raw object points in 3D space. Those points are in the cutting plane through the relevant vector point, when they are under a given distance threshold. These skeletal points can be extracted from planar object. Compared with us, Tagliasacchi et. al minimize the sum of squared distances from the point to the related normals [144]. That method will cause the problem that the position of the skeletal points for the planar object is infinite. Afterwards, the best line l can be reached by minimizing the sum of distances with the extracted skeletal points. The line detection result is shown in Figure 5.2d).

5.3.3 *Skeleton Topology Extraction*

As shown in Figure 5.2d, the result of line detection does not constitute the entire skeleton of articulated object. It is because that some skeleton points have been filtered out by clustering step. To deal with this case, the line growth algorithm is used to estimate the entire skeleton topology. All of the detected lines grow in both positive and negative direction to overcome the object's skeleton. The lines will stop growing when they arrive in two situations as follows:

(i) reach the edge of the object point cloud and are viewed as skeleton root node as the Node 1 and Node 5 in Figure 5.3;

(ii) meet another skeleton line and at that time they stop growing up and are characterized as skeleton link node as the Node 2, 3 and 4 in Figure 5.3.

These points are clustered and merged by 3D Euclidean clustering [160]. After that, the entire object skeleton nodes are extracted. Simultaneously, the root and link nodes indicate the topology of the skeleton of articulated object. These results are shown in Figure 5.2e and Figure 5.4c. Different colored points represent the different estimated skeleton nodes. The dashed line links represent the skeleton topology.

5.3.4 *Determination for The Number of Kinematic Joint*

As shown in Figure 5.3 and Figure 5.4d, the object skeleton topology is extracted to represent the different configuration of articulated objects. The dashed lines represent the object skeleton topology and the traces of different extracted skeleton nodes are shown as different colored solid lines. With the traces of skeleton nodes in different frames, all of the dynamic observations are presented. From frame 1 to 8, it is obvious that the observation patterns of nodes 3 to 5 differ from the patterns from frame 8 to 11. These two types of pattern in terms of the skeleton topology of object are changing based on this analysis. It implies that the estimated object is not a single joint articulated object. The skeleton node S_i with index i is viewed as the base node to estimate the Euclidean distances with others as $E_i = \|S_0 - S_i\|, i \in [1, n]$. These distances are used to calculate the difference cost function DIF_j between current frame j with the previous frame $j-1$ as following:

$$DIF_j = \sum_{i=1}^{n} \frac{|E_i^j - E_i^{j-1}|}{E_i^{j-1}},\ j \in [1,\ F] \tag{5.4}$$

where F is the number of frames. At frame 9, DIF_9 increases significantly, which means this articulated object contains multi kinematic joints. In comparison, the door of car is the single joint articulated objects as shown in Figure 5.4d.

With the certification of the joint number from the object skeleton topology observations under different configurations, the kinematic joint type can be characterized and the joint can be localized into the object. For a single joint articulated object, the trajectory of one skeleton node can represent the articulated object's motion pattern. This data is also used for its kinematic joint characterization. In another case as multi joint articulated object, we need to analyze all trajectories of its skeleton nodes hierarchically to extract the properties of kinematic joints respectively.

5.4 ARTICULATED JOINT TYPE CHARACTERIZATION

The kinematic joints of articulated object are constrained into two certain types, as prismatic and revolute [140]. Given the 3D trajectories of the end-effector of the object, it is rather straightforward to discriminate as these two types of kinematic joint. The position vector of the point A of an articulated object which is moving in the 3D space can be expressed by

$$\vec{g} = g_x\hat{x} + g_y\hat{y} + g_z\hat{z}. \tag{5.5}$$

In the case that only one positional component is non-zero, the joint is characterized as prismatic one. The positional components are in need to be digitized as follows: if a component is different than zero, it is assigned as value 1, else value 0. The digitized components g_x, g_y and g_z are used as the input for a Boolean logic scheme which is equivalent to the numerical computation given by

$$Y = (g_x + g_y + g_z - g_x g_y g_z)\,(g_x + g_y - g_x g_y)\,. \tag{5.6}$$

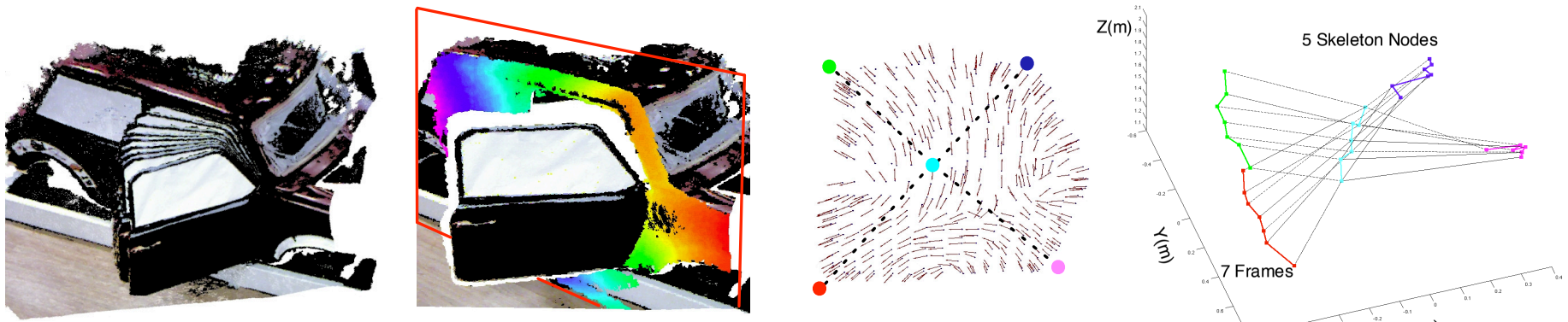
We apply Equation 5.6 at each time step and taking the average $\bar{Y}$ of all outputs $Y(n)$ where n is the time index. The results deduce whether the joint is revolute or prismatic. If $\bar{Y} = 0$, the joint is prismatic. If $\bar{Y} \neq 0$, the joint is revolute. In case that the joint is revolute which causes a rotational movement, the angle range of the joint is estimated as the constraint of working space. The positional data of the end-effector of an articulated object are recorded during demonstrations of the task. The angle range is computed by

$$\theta(n) = \arctan(\bar{g}_i(n)/\bar{g}_j(n)), \tag{5.7}$$

where $n = 1, ..., N$ is the time index and $\bar{g}_i$ and $\bar{g}_j$ the two non-zero average positional trajectories in directions i and j. The average positional trajectories are computed, since many demonstrations are available, as

$$\begin{aligned} \bar{g}_i(n) &= \frac{1}{K}\sum_{1}^{K} g_i^{(k)}, \\ \bar{g}_j(n) &= \frac{1}{k}\sum_{1}^{K} g_j^{(k)}, \end{aligned} \tag{5.8}$$

where $g_a^{(b)}$ is the position of demonstration b in direction a and K is the number of demonstrations of the task.



a) Data acquirement with different configurations

b) Object extraction based on RANSAC plane estimation

c) Vector field and extracted skeleton topology

d) Skeleton node trace through different frames

Figure 5.4: Skeletonization of a car door which has a single revolute kinematic joint.

5.5 LEARNING FORCE SKILLS FROM MANIPULATION OBSERVATION AND MAPPING

It is necessary to extract an average expert behavior for a task based on multiple demonstrations [85]. Since the speed of the demonstrator varies from trial to trial and demonstrations are not time-aligned, demonstrations become time-aligned by Dynamic Time Warping. The force policy of a task is extracted from multiple demonstrations using a probabilistic approach proposed in [20]. This approach consists of Gaussian Mixture Modeling and Regression. It can estimate a smooth generalized version of demonstrated signals which captures all the important features of the task.

Time-aligned data pairs $d_i = \{s_i, \mathbf{f}_i\},\ i = 1, ..., N$ are considered, where N is the number of data points in each demonstration, s_i the input joint states and $\mathbf{f}_i \in \mathfrak{R}^{D\times N}$ represent force data where D is the dimensionality of $\mathbf{f}$. A mixture of L Gaussian functions is considered with probability density function as

$$p(d_i) = \sum_{l=1}^{L} p(l)p(d_i|l), \tag{5.9}$$

where $p(d_i|l)$ is a conditional probability density function and $p(l) = \pi_l$ is the prior of the l-th distribution. We model the mapping from joint angles to endpoint forces by a mixture of L Gaussian functions. It is

$$p(d_i|l) = \frac{1}{\sqrt{(2\pi)^D|\Sigma_l|}} \exp(-\frac{1}{2}\left((\xi_i - \mu_l)^T\Sigma_l^{-1}(\xi_i - \mu_l)\right)) \tag{5.10}$$

where $\{\pi_l, \mu_l, \Sigma_l\}$ is the Gaussian function's parameter set represented by the prior probability, the mean and covariance matrix. The parameters of the mixture are estimated using the Expectation-Maximization (EM) algorithm. Following learning of the mixture parameters, a generic form of the signals $\mathbf{f}_i$ is reconstructed using Gaussian Mixture Regression (GMR). The states s_i are employed as inputs and the output vectors $\hat{\mathbf{f}}_i$ are estimated by regression. The mean and covariance matrix of the l-th Gaussian component are defined as

$$\mu_l = \{\mu_{s,l},\ \mu_{f,l}\}, \quad \Sigma_l = \begin{pmatrix} \Sigma_{s,l} & \Sigma_{sf,l} \\ \Sigma_{fs,l} & \Sigma_{f,l} \end{pmatrix}. \tag{5.11}$$

The conditional expectation and covariance of the signal $\mathbf{f}_l$ given s are

$$\begin{aligned} \hat{\mathbf{f}}_l &= \mu_{f,l} + \Sigma_{fs,l}(\Sigma_{s,l})^{-1}(s - \mu_{s,l}), \\ \hat{\Sigma}_{f,l} &= \Sigma_{f,l} - \Sigma_{fs,l}(\Sigma_{s,l})^{-1}\Sigma_{sf,l}. \end{aligned} \tag{5.12}$$

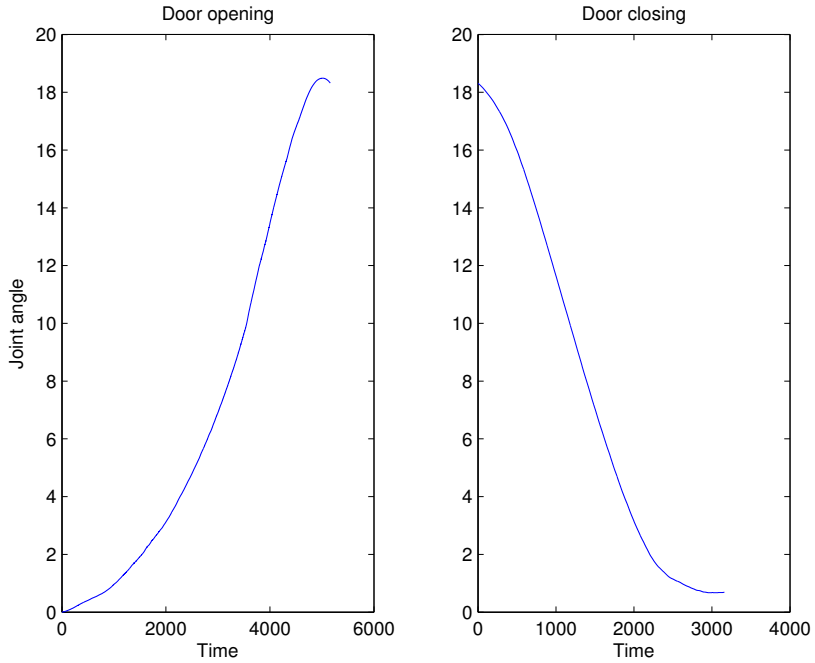


Figure 5.5: Angle state space estimated based on the position of the car's door handle. The joint angles are expressed in degrees. The time step is equal to 1ms.

Finally, the conditional expectation and covariance of $\mathbf{f}$ given s for a mixture of K Gaussian components are defined by

$$\begin{aligned} \hat{f} &= \sum_{l=1}^{L} \beta_l \hat{f}_l, \\ \hat{\Sigma}_{\mathbf{f}} &= \Sigma_{l=1}^{L}\ \beta_l^2 \hat{\Sigma}_{f,l}, \end{aligned} \tag{5.13}$$

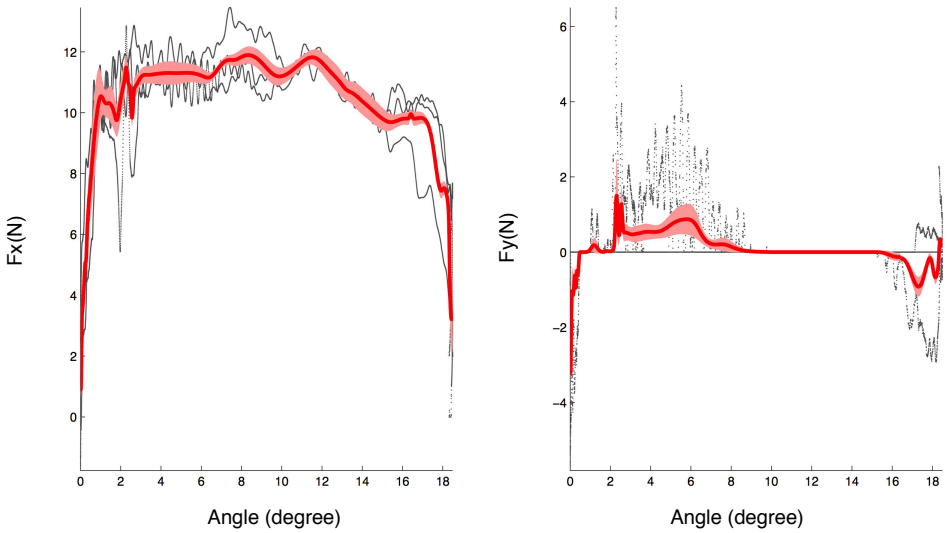
where $\beta_l = p(s|l)/\sum_{j=1}^{L} p(s|j)$ is the responsibility of the l-th Gaussian for s_i. The task force profile $\mathbf{f}$ is learned in the joint space s which is represented by the angle θ.

5.6 EXPERIMENTAL RESULTS

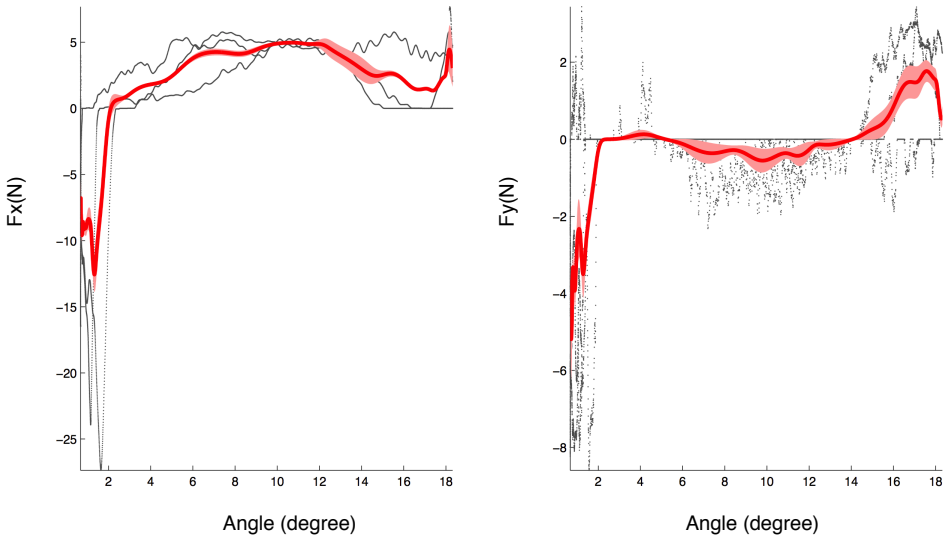
This work focuses on skeletonization and manipulating a single-joint articulated object. We demonstrate the performance of proposed method in a pitstop scenario where the single-joint car door is to be recognized and manipulated. A model of the door, represented by Equation 5.1, is built based on its skeleton topology, a kinematic descriptor of the joint and the end-effector force required for manipulation.

The point cloud of the door is acquired by one Kinect[1] sensor which is mounted on the top of an autonomous robot, as shown in Figure 2.4. This data is used for skeletonization of the door and estimation of the skeleton node traces over different

1 http://www.primesense.com



(a) Door Opening



(b) Door Closing

Figure 5.6: Learning the generalized 2-dimensional force profile of a task in joint angle space given 3 task demonstrations. (a) Door opening, (b) Door closing.

frames, as shown in Figure 5.4. The skeletonization of object is realized partially based on the Point Cloud Library[2]. We desire to learn manipulation skills in terms of the force which is required to open or close this single-joint car door.

Appropriate force is demonstrated to the robot by kinesthetic teaching and learned from multiple demonstrations of a task using the proposed approach. Several demonstrations of opening-and-closing task are provided to a 7 DoF robotic arm. Task space force as well as end-effector positional trajectories are captured during demonstrations. Following task space force learning, generalization is required to situations where the

2 http://www.pointclouds.org

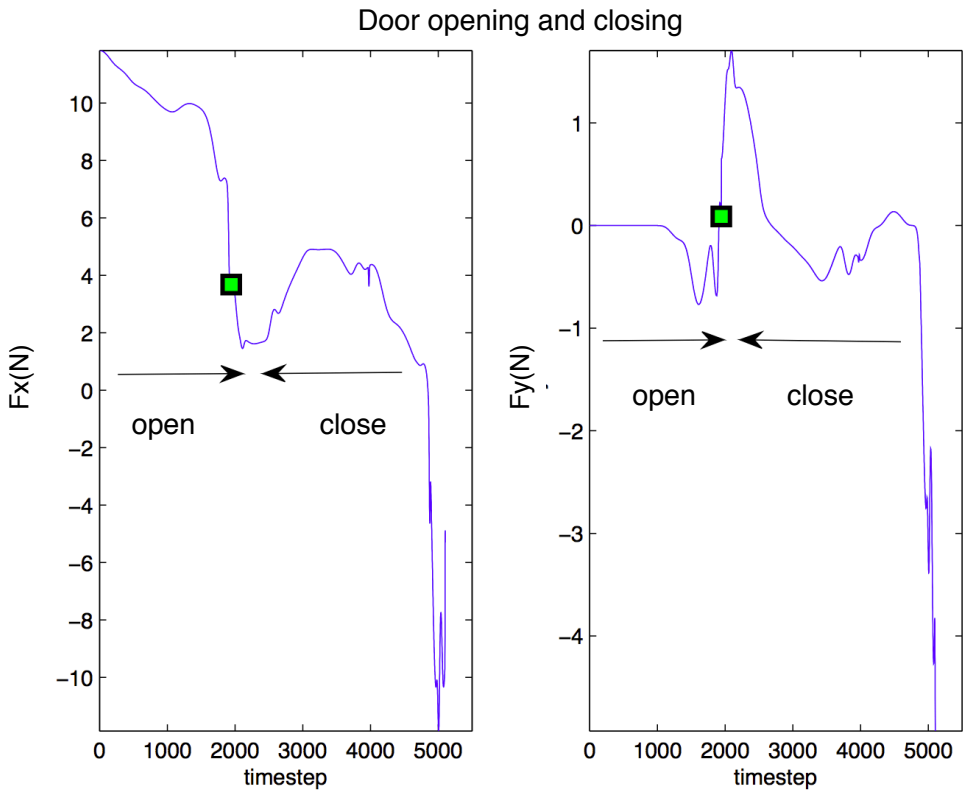


Figure 5.7: Door opening and closing where the door is initially open at 8 degrees. The time step is equal to 1ms.

initial door position may differ and based on the task goal such as the target angle of opening or closing. To accomplish this, the force constraints of the task are learned with respect to door's joint states. The current joint states are estimated using current frame's visual data.

Skeletonization of the car door is shown in Figure 5.4, where the door is recognized as single-joint articulated object using Equation 5.4. We observe that the trace of skeleton node has the same motion pattern with the robot arm end-effector trajectory. The current door's joint state can be achieved by the skeleton topology position and the learned door's rotational joint model.

Every demonstration consists of an opening and closing phase without any interruption between the two phases. The different start and end points of each trial are due to slight sliding movement of the robot end-effector along the handle of the door. Given manipulation trajectory, the type of joint is identified firstly by using the algorithm described in Section 5.4. This door's joint is characterized as revolute and estimate the joint space constrains which is computed, as shown in Figure 5.5. This angle space constitutes the input state space in terms of which the force trajectories are learned from multiple demonstrations.

Figure 5.6 shows the learning process of the 2-dimensional force for a door opening-closing task from 3 demonstrations by using the method described in Section 5.5. The force of robot's end-effector is learned separately for the two phases of the task. Following with the learning process, we desire to generalize the force generation policy to different tasks with different current state of articulated object. More specifically, the case is considered where the car door is already open at 8 degrees and the force profile is estimated which needs to be exerted in order to open the door completely and close it afterwards. Figure 5.7 shows the generalized force for this task where the two phases, opening and closing.

5.7 SUMMARY

In this chapter, we propose a novel method for modeling articulated object by combining visual and manipulation information. Visual processing contributes to recognize the object and identifying its structure and more specifically, its skeleton topology, the number and type of joints as well as the current joint states. Manipulation data represented by robot end-effector's force are learned from multiple task demonstrations in order to be able to operate the articulated mechanism. The forces are encoded with respect to joint states so that the system can generalize to new situations where the initial object configuration, and thus, joint state differs. The proposed method is demonstrated and validated in experiments as manipulation of a single-joint articulated. We bring this new idea for the visual representation of articulated object and also object's proper manipulation skills generation for autonomous robots.

6

REAL-TIME HUMAN BODY MOTION ESTIMATION BASED ON LAYERED LASER SCANS

In this chapter, we propose a method for real-time 3D human body motion estimation based on three-layer laser scans. All the useful scanned points containing human body contour information, are subtracted from the learned background of the environment. For human contour feature extraction. To avoid segmentation problems during human contour feature extraction, we propose a novel iterative template matching algorithm for segmentation and clustering. Robust distinct human motion features are extracted using maximum likelihood estimation and the nearest neighbor clustering method. Subsequently, the positions of human joints in 3D space are retrieved by associating the extracted features with a pre-defined articulated model of the human body. Finally we validate our proposed methods with experimental results, which show accurate human body motion tracking in real time.

The remainder of this chapter is organized as follows: Section 6.1 to Section 6.4 provide the detailed structure of the whole estimation system, including background subtraction, feature extraction, modeling of articulated human body and data association. The hardware setup and experimental results are presented and discussed in Section 6.5. Finally, Section 6.6 summarizes this work.

6.1 FRAMEWORK FOR REAL-TIME HUMAN MOTION ESTIMATION

In this section, we provide the details about the human body motion estimation system resulting in real-time estimation of body joint position in 3D space. The proposed method aims at solving the problem of acquiring the real human motion based on limited spatial data in terms of the scanned contour information retrieved from 3-layered laser scans.

The system is presented in Figure 6.1, where three LRFs are vertically aligned on different fixed heights from the ground plane. The heights are chosen in such a way that the LRFs can capture the arc-shaped contour points of the human's torso and upper arms, the hip and forearms, and the thighs separately. Foreground points are then extracted from the pre-learned background. During segmentation and clustering a novel iterative template matching method is proposed to solve the self occlusions and to get robust as well as distinct human motion features. The system is able to

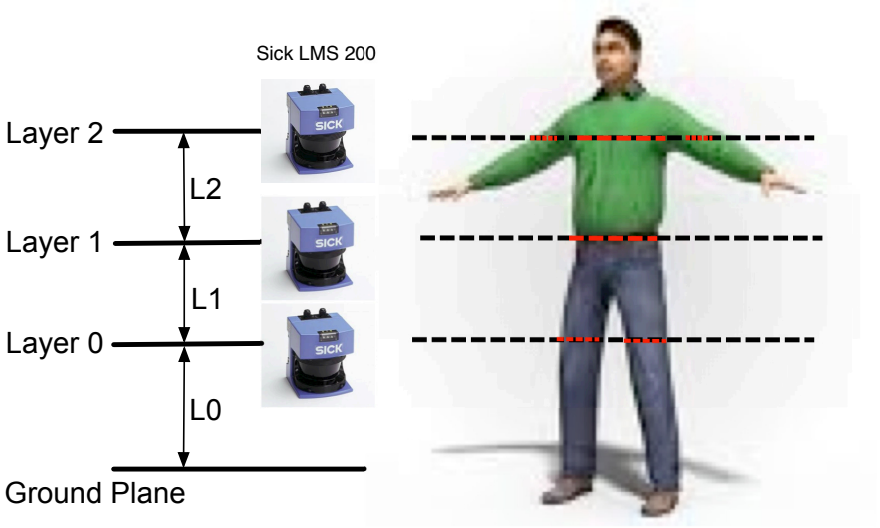


Figure 6.1: Overview of the proposed estimation system.

estimate the full human body motion after associating the extracted features with the pre-defined articulated human model in real-time.

The processing modules and components are illustrated in Figure 6.2. After the background subtraction in the raw data from each LRF is done, all the related human arc-shaped contour points are estimated. The human contour features are extracted using segmentation and clustering while the final pose is retrieved by associating these features with a pre-defined articulated human model.

6.2 HUMAN FOREGROUND DATA EXTRACTION

All the scanned contour points of objects in front of LRFs represent the raw data, which means they include both the background information (such as walls and other static objects) as well as moving object information. Background information subtraction is needed in order to extract the human contour information from the data.

The three LRFs measurements are with X and Y axes parallel to the ground plane and with heights z_0, z_1 and z_2. All scanned points can be represented as $P = \{p_i\}$ and $p_i = (x_i, y_i, z_i)$, where i is the point index.

The background is learned in the initial stage before the target enters in the scene. We take S scans to average the background information and save them in the background point set $P_b = \{\sum_{s=0}^{S} p_s/S\}$. Therefore, the foreground data P_f can be extracted by comparing the raw data with P_b with a given threshold, as shown in Figure 6.3.

However, it is possible that the background changes during the observation. This can happen for example when a piece of furniture is moved. In our strategy, we deal with

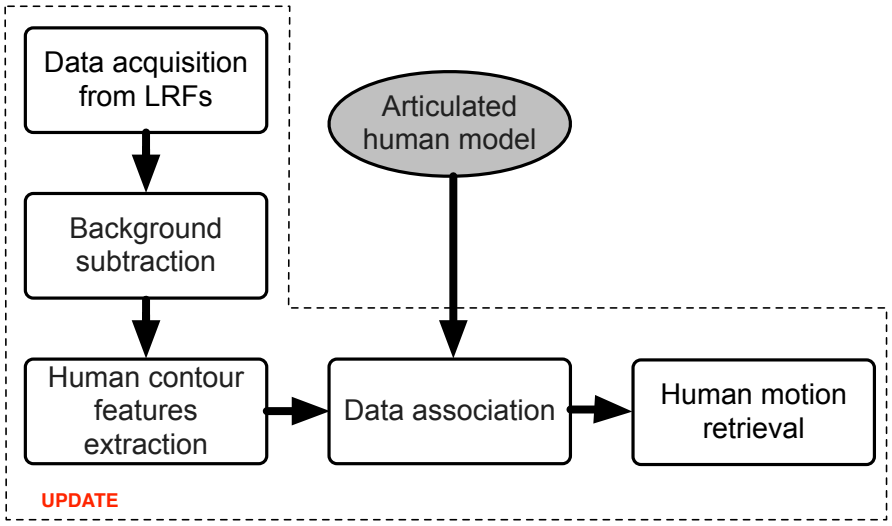


Figure 6.2: Diagram of the proposed estimation system.

this case by switching foreground points into background points when their position variations are under a given threshold [19].

6.3 HUMAN CONTOUR FEATURES EXTRACTION

From the foreground data P_f, useful features can be extracted which are represented as $F(c, \theta, l)$, where c is the cluster center position, θ is the rotation angle and l is the length of cluster. The information c and θ will be used to get the pose and l will be used to classify these human body parts. Using nearest neighbor and template matching techniques these two kinds of segmentation and clustering methods are used to obtain the needed features.

6.3.1 *Segmentation by Nearest Neighbor Clustering*

This segmentation criterion is based on the geometry relationship between the nearest neighbor points [35]. All the points that are within a predetermined distance are segmented as one cluster. The clusters that satisfy certain conditions will be viewed as the effective human contour features.

The algorithm for cluster segmentation is as follows:

1. Compute the distances D between each pair of consecutive points from the effective human data $P_h = \{p_i\}$ in LRF data image: $D_i = \|p_i - p_{i-1}\|$.

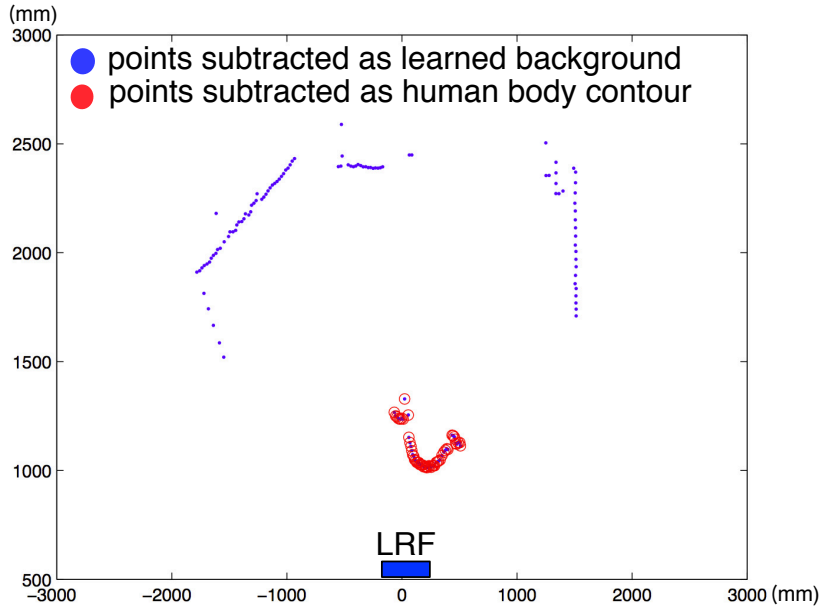


Figure 6.3: Extraction of human contour points from background on one layer laser scans.

2. Classify the suitable points into one cluster $C_j(P, n)$ with vector of points P and the number of points n using the distance threshold T_c: push p_i into C_j if $D_i < T_c$ otherwise create a new cluster.

3. Delete the cluster C_j with index j if its number of points n_{C_j} is under a number threshold T_n as $n_{C_j} < T_n$.

4. Compute the center position C_j as feature F_k position information:

$$c_{F_k} = (\sum x_{C_j}/n_{C_j}, \sum y_{C_j}/n_{C_j}). \tag{6.1}$$

5. Compute the rotation and the length of F_k based on the start- and endpoint (p_S, p_E) of the cluster C_j:

$$\begin{aligned} \theta_{F_k} &= \arctan \frac{y_{p_S} - y_{p_E}}{x_{p_S} - x_{p_E}}, \\ l_{F_k} &= \|p_S - p_E\|. \end{aligned} \tag{6.2}$$

However, this geometric clustering method fails when extracting the valid features in two typical situations as shown in Figure 6.4:

A. Occlusion appears and occluded template cannot be estimated from available estimation, e.g. forearm separates hip part cluster;

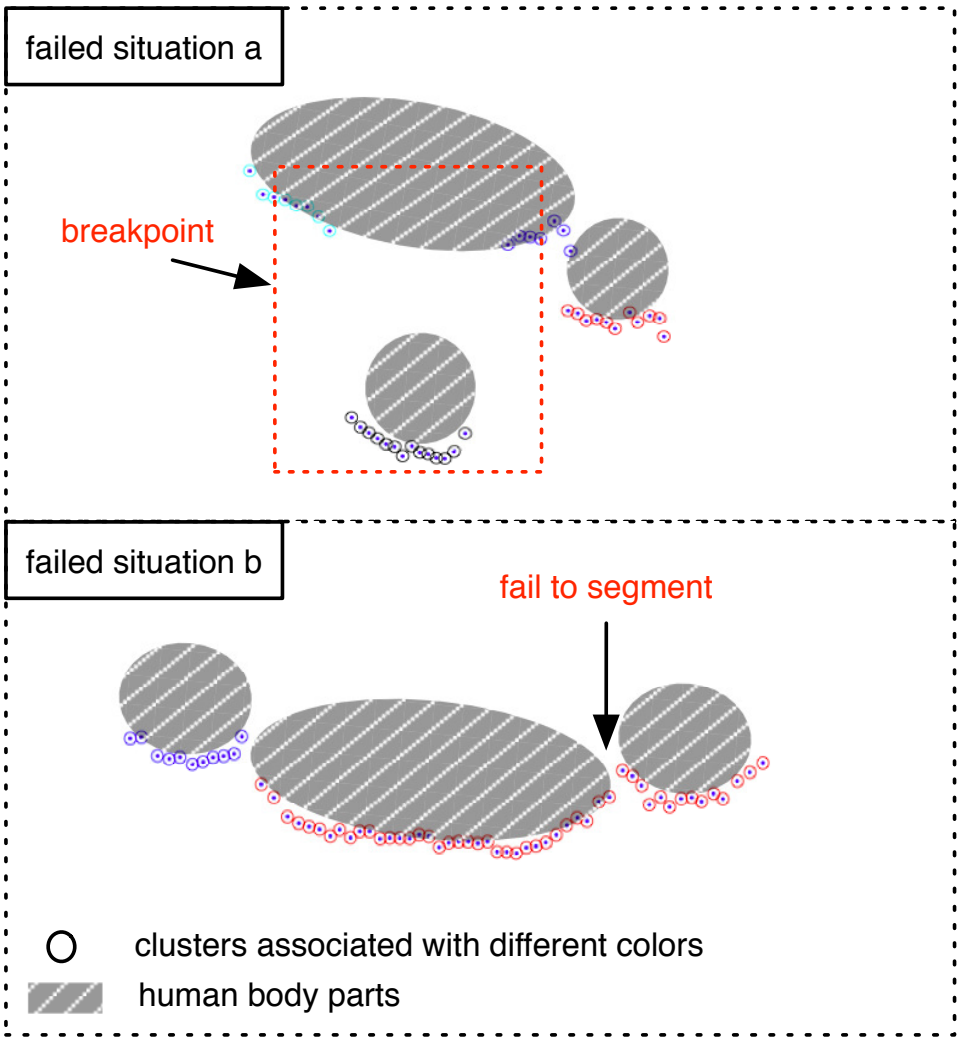


Figure 6.4: Two failure situations using NNC. Different colored points show distinct associated clusters.

B. The outliers of the template cannot be excluded which will influence the template estimation result, e.g. when upper arm is within distance threshold T_c to torso.

The final result generated from those estimated clusters in such situations will contain incorrect features. Consequently, template matching is considered to solve the problem of segmentation and associate the related information for clustering.

6.3.2 *Segmentation Using Template Matching*

Aiming to avoid the above failure situations, a circle template matching algorithm based on [145] is employed with the following assumptions: 1) the torso and hip contour are always scanned; 2) the shapes of torso and hip sections in terms of the contour information are not changing.

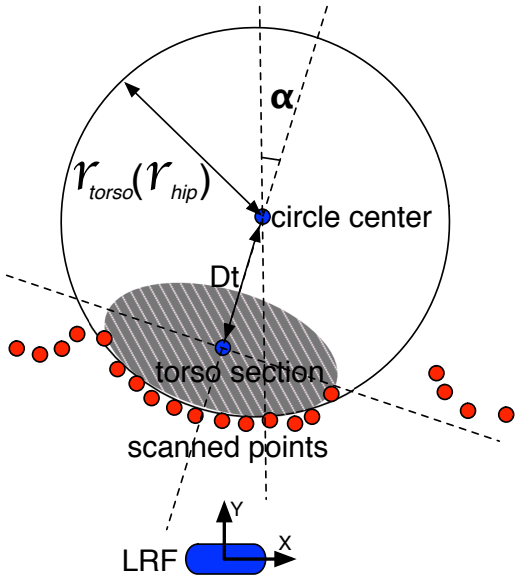


Figure 6.5: Torso circle template construction. The red points are scanned points. The real position of torso is represented by the shadow ellipse. The center of the torso can be obtained from the matched circle center position, rotation angle α and the constant distance parameter D_t.

The torso and hip circle template models are built, with each template having a different radius r_{torso} and r_{hip}. The center of the circle has a constant distance D_t to the center of the torso section, as shown in Figure 6.5. The rotation angle α can be obtained from the 5th step of the NNC algorithm in Section 6.3.1. The two radii of the circle models are defined as $r_{torso} = 320$ mm and $r_{hip} = 300$ mm respectively.

If the scanned points match the circle template, they need to fulfill the following equation

$$Dist = \sqrt{(x - x^*)^2 + (y - y^*)^2} - r = 0. \tag{6.3}$$

where (x^*, y^*) is the center of the circle template, r is the radius and $Dist$ is the distance between the point and its nearest circle border.

In order to obtain the center position of torso, the circle center needs to be computed first. Therefore, Maximum Likelihood Estimation (MLE) is applied here to estimate the center position of circle. Each scanned 2D laser point is assumed as having an independent error which can be represented as a Gaussian distribution with zero mean and standard deviation σ.

As $(\overline{x}_i, \overline{y}_i)$ is the true position of the scanned position (x_i, y_i) and n is the number of scanned points used to match the circle template, then the likelihood of all the points can be represented as

$$L(x, y) = \prod_{i=1}^{n} \left(\frac{e^{-\frac{(x_i - \overline{x}_i)^2}{2\sigma^2}}}{\sqrt{2\pi\sigma^2}} \frac{e^{-\frac{(y_i - \overline{y}_i)^2}{2\sigma^2}}}{\sqrt{2\pi\sigma^2}} \right). \tag{6.4}$$

For easier computation, we use $-\log L$ to minimize as

$$L_{-log} = -\log\left(\frac{1}{(2\pi\sigma^2)^n} e^{-\frac{\sum_{i=1}^{n} [(x_i - \overline{x}_i)^2 + (y_i - \overline{y}_i)^2]}{2\sigma^2}} \right). \tag{6.5}$$

By removing all the constants, which do not contribute to minimization, we obtain an equivalent formulation optimization problem with the modified cost function

$$L_{circle} = \sum_{i=1}^{n} \frac{(x_i - \overline{x}_i)^2 + (y_i - \overline{y}_i)^2}{\sigma^2}. \tag{6.6}$$

The Lagrange method of undetermined multipliers are used including the equality constraint given by Equation 6.3. The final equation to be used to get the MLE result is

$$L_{circle} = \sum_{i=1}^{n} \frac{[(x_i - x^*)^2 + (y_i - y^*)^2 - r^2]^2}{(x_i - x^*)^2 + (y_i - y^*)^2}. \tag{6.7}$$

Since the r is the parameter of circle template, MLE becomes the nonlinear problem of obtaining the parameters (x^*, y^*) for minimizing Equation 6.7. The Newton-Raphson (NR) method is adopted here to solve the optimization problem numerically.

6.3.3 *Iterative Template Matching for Segmentation and Clustering*

In addition, in order to obtain stable and accurate features, we propose an Iterative Template Matching for Segmentation and Clustering (ITMC) method here. This method can estimate the known template and other clusters whenever an occlusion happens. The pseudocode for ITMC is listed in Algorithm 6.1.

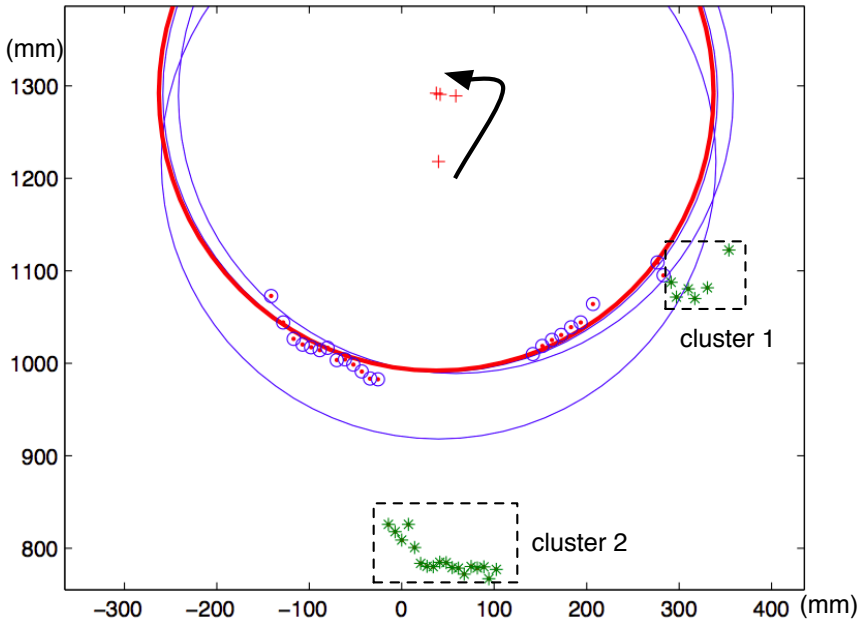
The main idea of ITMC is to update the input data every time to match the circle and exclude the points which are not related to the circle template. When the position of the matched circle becomes stable, NNC is applied to segment and cluster the excluded points. This algorithm can greatly improve the accuracy of segmentation and clustering results while also solving the problem of failure situations such as those shown in Figure 6.6. Notice that ITMC is employed on layer 1 and layer 2 which used the torso and hip section template, whereas only NNC is applied on layer 0, since

Algorithm 6.1 Iterative template matching for segmentation and clustering

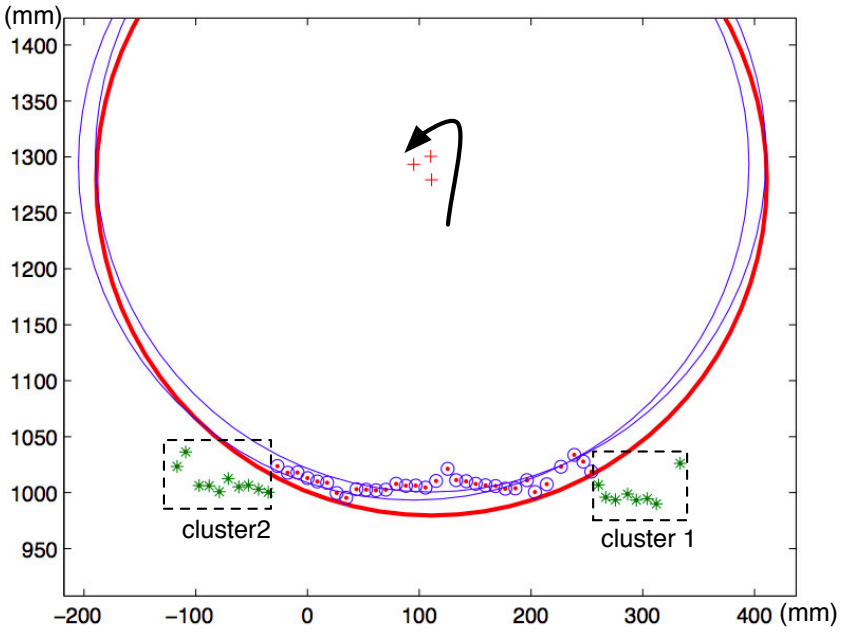
Int $i \leftarrow 0$; # iteration times for ITMC
Double $T_{outlier}$; # distance threshold to segment the outliers
Double ε_{itmc}; # threshold for ITMC
Double R_{circle}; # radius of circle template
Double $CE_0 \leftarrow (x_0^*, y_0^*)$; # initialize as center of point set P as $\{p_i\}$

# if error is larger than the threshold
while $\varepsilon \geqslant \varepsilon_{itmc}$ **do**
  $i++$; # increase the iteration times
  MLE $(P, CE_{i-1}, R_{circle})$; # MLE circle matching
  $CE_i^* \leftarrow (x_i^*, y_i^*)$ # get matched point's center
  for $j := 0$ **to** $P.pointsize$ **step** 1 **do**
    # each point's distance to the circle
    $D_j \leftarrow \text{abs}(\|P_j - CE_i\| - R_{circle})$;
    if $D_j > T_{outlier}$ **then**
      push P_j into O; # push into the outliers
    end if
  end for
  $P_m \leftarrow$ all points of O; # reduce the outliers to update the input data
  $P \leftarrow P - P_m$;
  $\varepsilon \leftarrow \|CE_i - CE_{i-1}\|$; # displacement of estimated circle position
end while

NNC$(O, CE_i) + cluster\{P\}$; # segment and cluster based on NNC
$C \leftarrow \{C_i\}$



a) segment and cluster the hip part



b) segment and cluster the torso part

Figure 6.6: Clustering based on ITMC. The final matched circle is red. A black arrow shows the trace of the center of the matched circle during the iterative steps. (a) use 4 times iteration to get the center of the template of the human hip circle, meanwhile segment the forearm clusters; (b) iterate 3 times to get the center of the template of the human torso circle, meanwhile segment the upper arm clusters.

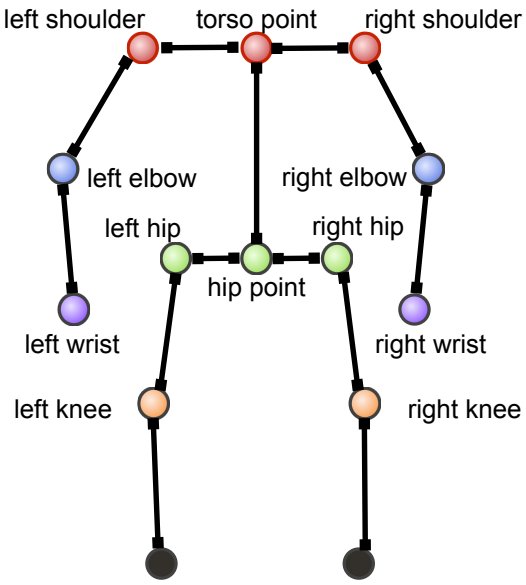


Figure 6.7: Articulated human model for data association.

Table 6.1: Parameters of human model.

Link	Start Joint	End Joint	Length(mm)	DOFs
1	hip point	torso point	490	3
2	torso point	left shoulder	210	1
3	torso point	right shoulder	210	1
4	left shoulder	left elbow	290	3
5	right shoulder	right elbow	290	3
6	left elbow	left wrist	320	3
7	right elbow	right wrist	320	3
8	hip point	left hip	200	1
9	hip point	right hip	200	1
10	left hip	left knee	500	3
11	right hip	right knee	500	3

it can achieve these clusters for legs feature extraction successfully. Subsequently, all human body features are extracted in real time based on data of the three LRFs.

6.4 HUMAN MODELING AND DATA ASSOCIATION

6.4.1 *Articulated Human Model Building*

The above extracted features $F(c, \theta, l)$ represent the 2D data at different heights. As the three layered lasers have a fixed height above the ground plane, the system is able

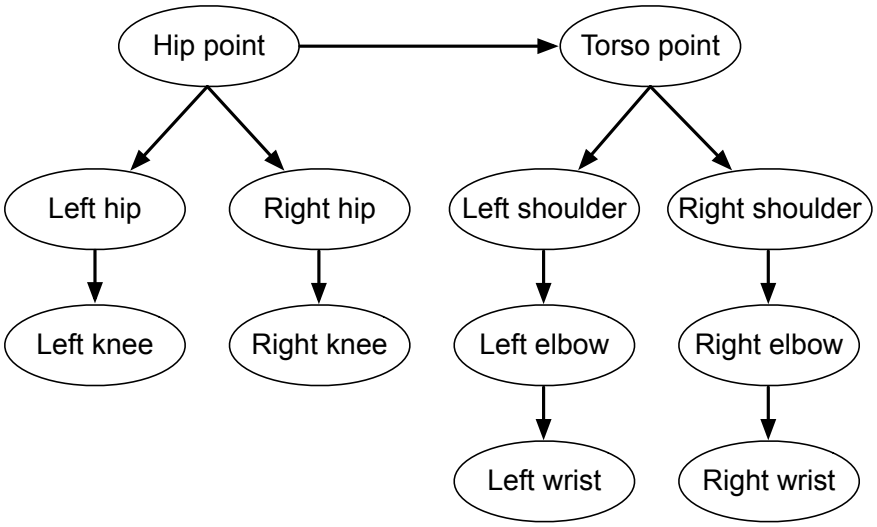


Figure 6.8: Hierarchical computation for each joint

to estimate a particular person. To get 3D human joints data, the features need to be associated with the pre-defined articulated human model (illustrated in Figure 6.7). This model has 11 fixed length links with 25 degrees of freedom. Details of the pre-defined human model are shown in Table 6.1. These coefficients and the radii of torso and hip templates mentioned in Section 6.6 are measured from the people proposed as motion estimation targets.

6.4.2 *Contour Feature Association with Human Model*

The human contour features f_{torso} and f_{hip} are obtained using the proposed ITMC approach. All other features are based on the position relationship between layers, in particular, right and left arms are classified depending on the positions of cluster centers. Classified features can then be associated with the corresponding parts of the human body such as:

$$\begin{aligned} F_{body} = \{&f_{torso}, f_{hip}, f_{l-upperarm}, f_{r-upperarm}, \\ &f_{l-forearm}, f_{r-forearm}, f_{l-thigh}, f_{r-thigh}\}. \end{aligned} \tag{6.8}$$

Each feature just represents the horizontal sectional center position of the associated human body part. Note that the position data of F_{body} is a 2D position with fixed height. Consequently, the next step is to get each related human joint position in 3D space based on these extracted features and retrieve the human body motion in real time.

In order to simplify the association of features, the height of hip in terms of $f_{l-thigh}$ and $f_{r-thigh}$ is fixed, which is a reasonable approximation as long as the human is

standing or walking only [55]. Furthermore the links 2, 3, 8 and 9 are approximated to have only one DOF, as we ignore the vertical rotation of the torso and hip. In addition, the foot joints are also neglected because of the lack of related height information. With these assumptions, all associated human body part features F_{body} are extended into related human joint data which are hierarchically computated as shown in Figure 6.8:

1. Based on the torso and hip features, get the hip point with the fixed height;
2. Find the torso joint based on the fixed length of torso and direction with torso and hip features;
3. Compute the rest of the joints hierarchically.

Finally, all predefined human joint positions in 3D space can be estimated as P_{joints}. The position data are then filtered using a Kalman filter to get a smoother result.

6.5 EXPERIMENTAL RESULTS

The experimental setup consists of 3 SICK LMS-200 laser range scanners from SICK AG[1]. The set of LRFs at each layer can scan with distance resolution of 10 mm and angular resolution of 0.5 degree, angular range of 180 degree, and data transmission rate at 500 kBps using the RS-422 interface. In the experiments, the three LRFs have been fixed to a height of 590 mm, 950 mm and 1255 mm with respect to the ground plane. The synchronization of data from the LRFs is not a problem due to the relatively fast scanning rate while careful mounting also alleviates the need for additional calibration. We use the Sick LIDAR Toolbox and the experiments to process for real-time humane body motion estimation utilizes Matlab on a Core Duo PC (Linux kernel x86). The entire time of processing is below 40 ms, which is fast enough to estimate human motion at real time. In our experiments, multiple types of human motion are considered which include several typical behavior patterns, such as walking, running, arm waving and moving sideways while standing in front of the multi-layer lasers as our sensor system. Currently, our proposed system has been tested on the single specific human pose estimation. It is due to the particular human model and the limitation of the source data from lasers with the fixed heights.

In order to effectively evaluate our proposed system, a Kinect is utilized as a reference to compare based on the estimation results of "human skeleton tracking" interface in the Kinect middleware. Figure 6.9 shows the experimental results of our proposed 3-layer LRF estimation system as well as the estimation results of tracking system provided by Kinect. These two systems are tested at the same environment and at the

1 http://www.sick.com

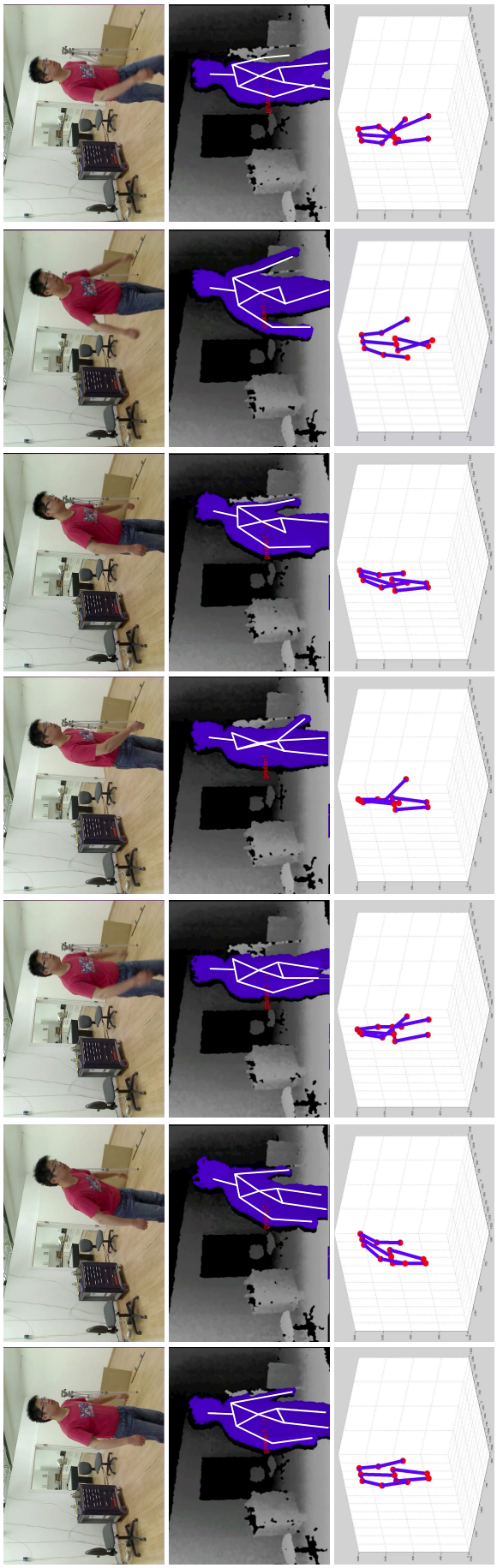


Figure 6.9: Human body motion estimation using OpenNI user tracker from Kinect (middle figure) and using the proposed 3-layer LRFs (bottom row). The top row shows the image reference.

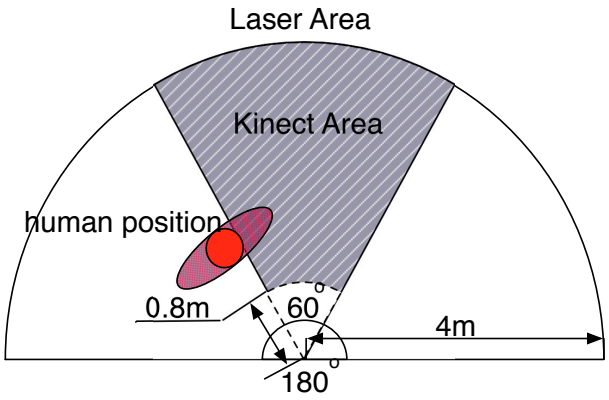


Figure 6.10: Different effective areas of the Kinect sensor and the laser scanners.

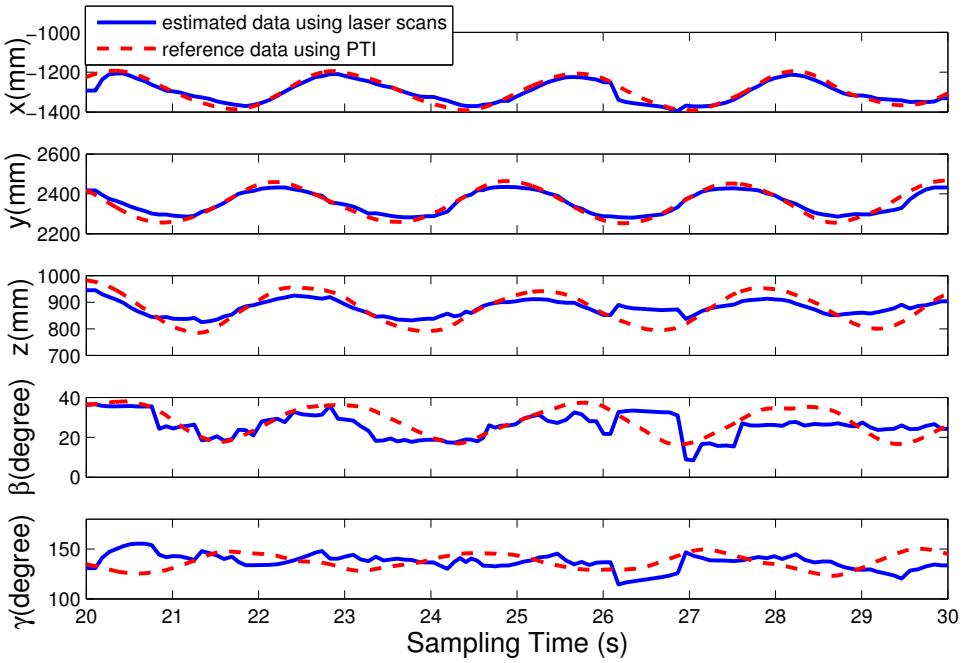


Figure 6.11: Measured and estimated pose of right arm during dynamic motion-back-forth swinging of the arm. Pose is described with wrist position (x, y, z) , the angle β and the angle γ. Estimated data using laser scans is shown with full blue line and reference data using PTI is shown with red dashed line.

same time. The results of comparison show that, our proposed system achieves accurate full body motion, while in the effective range of the Kinect. The system performs very close to the results obtained by the interface of Kinect. However. note that Kinect sensor's angular range is only 60 degrees, while the 3-layer LRF features a full 180 degrees. As shown in Figure 6.10, our estimation system still can perform quite well in the location with larger angle, while the Kinect sensor does not provide any results with its limitation of working space.

Table 6.2: Root mean square error for the right arm pose

position error (mm)			angle error(degree)	
x	y	z	β	γ
21.63	22.79	33.43	6.12	11.84

Furthermore, the processing speed of our proposed system is 25 Hz. And it is superior to Kinect's 13 Hz due to its high computational cost. Moreover, the Kinect sensor does not work in outside environment which constrains robot's working area. That is also the benefits that our sensor system is suitable and reliable for the applications of autonomous robot which is working outside, such as our ACE city explorer robot. All in all, despite of the fewer data information of LRF, a relatively good estimation result can be obtained within the area of half circle (180 degrees) with 4 m radius. For the data accuracy evaluation, a Phoenix Technologies Incorporated VZ-4000 3D position motion sensing system (PTI)[2] is used to capture the joint's position as ground truth. Because of the limitation of the working area of PTI tracking system, only the right arm's motion are analyzed here, which is represented by the right wrist's 3D position $p_{r-wrist} = (x, y, z)$, the angle β between the torso and upper arm and the angle γ between upper arm and forearm. This data can comprehensively describe the real pose of human's right arm and represents the motion of relevant human joints. The estimation is evaluated during human's movements. The accuracies of position and angle are shown in Figure 6.11. More precisely, Table 6.2 illustrates the evaluation of root mean square deviation (RMSD) during the experiment. From these evaluations, the estimated arm pose is very accurate compared with the measured tracking result from PTI system as ground truth. Nevertheless, the errors of z direction and in the angles of arm are relatively larger, which is mainly ascribed to the redundancy and the imprecise parameters of the pre-defined articulated human model. From Figure 6.11, the incidental jumps of the estimated tracking result are due to the result of occasional mismatching of human contour features from the scanning.

6.6 SUMMARY

In this chapter, we propose a real-time human body motion estimation system based on 3-layer laser range finder scans. This system is singularly appropriate to autonomous robots which explore the outdoor environment. The estimated human body motion provides the potentials for human robot interactions. The three layered lasers provide the 2D points at different hight respect to the ground plane. This data represents the

2 http://www.ptiphoenix.com/

human contour information and is extracted from the learned background environment at first. Afterwards, we used iterative template matching for segmentation and clustering method (ITMC) to get robust extraction of body parts. This novel method can solve the problems of segmentation and clustering in some cased during the movements of human. From experimental results, our approach can simultaneously retrieve the human motion and the accurate 3D position of human joints in real time.

7

CONCLUSIONS AND FUTURE WORK

In this final chapter, we will summarize the main contributions of this thesis and thereby concluded these works. According to these observations, possible improvements and some interesting directions for future research are pointed out.

7.1 CONCLUSIONS

This thesis has contributions on the scene understanding and recognition from visual data structure properties in dynamic environment. We analyze and extract the effective and comprehensive information from the analysis at different levels of visual data structure. Based on these, we are able to to achieve the understanding and recognition results, such as rigid or deformable object identification, pose localization and motion observations et al.. These visual data structures are constructed hierarchically at different-levels as "iconic images", "segmented images", "geometric representations" and "relational models". The raw visual data of the perception system is epresented as RGB images, depth images and multi-layer 2D points from multiple types of sensors. That gives huge potential possibilities for different applications in the research and industrial domains. We also applied various contributions to related computer vision and robotics applications of texture/textureless rigid object recognition, dense/deformable motion extraction for dynamic scenes, articulated object recognition and manipulation, and real-time human body motion estimation from multi-layer laser scans. We will present the detailed contributions and conclusions of these works separately as follows:

3D Object Recognition and Pose Estimation: We proposed a novel global object descriptor, so called Viewpoint oriented Color-Shape Histogram, which combines 3D object's color and shape features. The descriptor is efficiently used in a real-time textured/textureless object recognition and 6D pose estimation system, while also applied for object localization in a coherent semantic map. We initially build the object model by registering multi-view color point clouds, and generate partial-view object color point clouds from different synthetic viewpoints. Thereafter, the extracted color and shape features are correlated as a VCSH to represent the corresponding object patch data. For object recognition, the object can be identified and its initial pose is estimated through matching within our self created database. Afterwards the object pose can be optimized by utilizing an iterative closest point strategy. Therefore, all the objects in

the observed area are finally recognized and their corresponding accurate poses are retrieved. We validate our approach through a large number of experiments, including daily complex scenarios and indoor semantic mapping. Our method is proven to be efficient by guaranteeing high object recognition rate, accurate pose estimation result as well as exhibiting the capability of dealing with environmental illumination changes.

Dense and Deformable Motion Estimation: We presented a novel hierarchical MRFs optimization method for dense and deformable motion extraction in dynamic RGB-D scenes. In particular, this hierarchical MRFs structure consists of two layers, respectively named the segmentation and correspondence layer. Firstly, in the segmentation layer, the dynamic foreground data is successfully segmented through a pixel-level MRF. Secondly, in the correspondence layer, the extracted foreground data is structured as a 3D point-level MRF. A new surface descriptor named the "deformable color and shape histogram" is proposed. It is combined with photometric and geometric features to represent a deformable surface. The foreground data correspondences across consecutive frames are extracted next. Finally, the dynamic scene motion is retrieved correctly from these correspondences. The discrete optimization scheme is utilized for these binary classification and multi-labeling problems. Moreover, a dataset of dynamic RGB-D scenes is built, which involves different motion patterns and surface properties of dynamic foreground. The effectiveness and efficiency of our proposed approach for high accurate foreground segmentation and motion extraction is validated in experiments.

Articulated Object Recognition and Manipulation: We proposed an approach to model articulated objects by integrating visual and manipulation information. Line-shaped skeletonization based on depth image data is realized to extract the skeleton of an object given different configurations. Using observations of the extracted object's skeleton topology, the kinematic joints of the object are characterized and localized. Robot end effector's force data in the form of task-space force required to manipulate the object, are collected by kinesthetic teaching and learned by Gaussian Mixture Regression in object joint state space. Following modeling, manipulation of the object is realized by first identifying the current object joint states from visual observations and second generalizing learned force to accomplish the new task.

Real-Time Human Motion Estimation: We proposed a method for real-time 3D human body motion estimation based on three-layer laser scans. All the useful scanned points, presenting the human body contour information, are subtracted from the learned background of the environment. For human contour feature extraction, in order to avoid the situations of unsuccessful segmentation, we propose a novel iterative template matching algorithm for clustering, where the templates of torso and hip sections are modeled with different radii. Robust distinct human motion features are extracted using maximum likelihood estimation and nearest neighbor clustering method. Sub-

sequently, the positions of human joints in 3D space are retrieved by associating the extracted features with a pre-defined articulated model of human body. Finally we demonstrate our proposed methods through experiments, which show accurate human body motion tracking in real time.

7.2 FUTURE WORKS

Although the results presented in this thesis are quite promising, there still exists several open research challenges that remain for future investigation.

From research direction on the analysis of the visual data structure representation, currently most of the focus is on image's inlier neighborhood relationship in both photometric and geometrical aspects. Hence, the resolution and quality of the sampled points have the largest influence on the feature point extraction and descriptor generation, so that further analysis of the properties of such methods are warranted. One of further roads would be inspired by human perception and cognition to deal with the noisy data, low resolution input and special cases involving missing data. That would be interesting to utilize the data structure analysis directly for the raw sensor data and reconstruct the data afterwards. On the other hand, from our understanding, it would be an interest research direction to put more weights on the dynamic structural components into the visual data analysis for scene understanding and recognition, such as object physical model and motion constrains. That has the potential for improved spatial and temporal visual feature extraction for detection, recognition and tracking.

We have contributed in multiple vision problems which appear in our daily life. For future experimental-oriented applications, there are also a lot of potential for adapting our work to robotics and industrial domains.

Regarding the topic on rigid recognition and pose estimation, it would be interesting to extend the work onto the part of system optimization and model building of wider-variety of objects. We have currently collected a variety of object model dataset from our modeling platform and also third-party modeling approach, such as CAD modeling. We would like to apply our system into the industrial cooperations. Especially the automatic production line's quality testing and pickup application for industrial robotic arms. Our system can recognize the object within all possible 6D poses, which also can be retrieved for next-step applications. In the domain of robotics research, we already established the successful integration in robotic navigation and object exploration applications in large-scale semantic maps. It combined the general semantics and also the detailed object informations for the object-oriented scenarios. That would be also an interesting direction for robotic grasping and manipulation. Since we already have the 3D object model, it is possible to generate the best grasping points for

a specific object while also considering manipulation skills. To reach this goal, a probabilistic map needs to be derived from 3D space analysis from the estimated objects and complex background environment. Additionally, we would like to mention that it is planned to release our object model dataset and source code to the ROS community.

In the application of dynamic motion extraction, we focused on dynamic motion extraction only from the visual data structure with spatial and temporal relationships. The GPU-based optimization is in plan to speed up the correspondence process for real-time scenarios and for the construction of datasets consisting of a wider-variety of dynamic scene sequences. In the near future, we plan to release this dataset into public with ground truth information on the motion. Our system has no motion constrains or any deformation assumptions that limit its application to specific computer vision scenarios. With the solution on this general problem, there exist a lot of future extensions that would enable a multitude of computer vision and robotics applications. For instance, one direction is the rigid part segmentation, with the extracted motion field of the entire dynamic scene sequence. It would be possible to segment the relative rigid parts from the motion properties. Another application would be articulated object modeling from their motion field. It provide the potential to classify different joint types, to estimate the object joint space working constraints and to map the joint information into articulated objects. Moreover, we are testing to apply our hierarchical MRF into the medical image registration and disease diagnose from the estimated motion field.

For the articulated object recognition and manipulation applications, we currently focus only on single-joint articulated object currently. To the best of our knowledge, we are the first to propose an articulated object modeling method that combines visual and manipulation observations. This new method can be applied into different robotic applications. Future work will focus on modeling of a wide-variety of objects which also involve more than one joint. There exist several potential extension of our method. In particular, our object skeletonization method can be employed as part of a robot arm configuration and calibration scheme. Because of the limitation that we only get the arc-shaped object surface information from single Kinect mounted on autonomous robot. As the initial step we are implementing the object shape completion process in order to extract an accurate skeleton of the articulated object for best representing and recognizing the object. Moreover, another direction is to use the extracted motion field based on our dynamic dense/deformable scene motion estimation system, which can be directly adapted for the articulated object modeling.

Based on our contributions in real-time motion extraction, we can apply this method for human robot interaction applications. Currently, we have tested the single person motion estimation. The data fusion and association with human model are developed to retrieve the real-time motion, based on limited multi-layer 2D scanned points. Fu-

ture work will focus on the multiple person motion estimation with different articulated human models. Because of the limitation of current sensor technology, some sensors like Kinect and Time-of-Flight camera would not be considered to utilize for autonomous robots that is working outside. However, laser range finder as a suitable sensor, it only provides limited information that cannot represent the real 3D environment correctly. Hence, it is considered to develop our system to initially fit a different person using our automatic human model building implementation. Moreover, it is possible to extend our work into the pose classification based on feature learning and recognition methods. To extract more detailed human data, it is imaginable to experiment with 3D lidar scanners, such the velodyne lidar scanner used in the Google Car project[1]. We are considering to use a panning or tilting laser to get the 3D environment in front of the robot. Nevertheless, it leads relative lower frequency, which depends on the speed of actuator. To deal with that, it is possible to extract the useful information as interests utilizing optimization schemes, such as attention control in active vision applications.

To conclude, this thesis has presented our research on dynamic scene understanding from properties of visual data structures. Our method is contributed and validated from the innovations which are approached to solve different visual analysis and robotic vision problems. We believe that the achievements of this research has huge possibilities to be extended in large variety of application in near future. Additionally, we hope that our work increase the dependability, flexibility, and ease of usage of visual data structure analysis for vision and robotics, thereby contributes to development of further research work and industrial applications.

1 thttps://plus.google.com/+GoogleSelfDrivingCars/videos

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

BIBLIOGRAPHY

[1] A. E. Abdel-Hakim and A. A. Farag. Csift: A sift descriptor with color invariant characteristics. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1978–1983, 2006.

[2] A. Agarwal and B. Triggs. A local basis representation for estimating human pose from cluttered images. In *Proc. of Asian conference on Computer Vision (ACCV)*, pages 50–59, 2006.

[3] M. Agrawal, K. Konolige, and R. C. Bolles. Localization and mapping for autonomous navigation in outdoor terrains : A stereo vision approach localization and mapping for autonomous navigation in outdoor terrains : A stereo vision approach. In *Proc. of IEEE Workshop on Applications of Computer Vision*, pages 7–12, 2007.

[4] N. Ahmed, C. Theobalt, C. Rossl, S. Thrun, and H.-P. Seidel. Dense correspondence finding for parametrization-free animation reconstruction from video. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008.

[5] C. B. Akgül, B. Sankur, F. Schmitt, and Y. Yemez. Multivariate density-based 3d shape descriptors. In *Proc. of IEEE International Conference on Shape Modeling and Applications*, pages 3–12, 2007.

[6] M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. T. Silva. Computing and rendering point set surfaces. *IEEE Transaction on Visualization and Computer Graphics*, 9(1):3–15, 2003.

[7] K. S. Arun, T. S. Huang, and S. D. Blostein. Least-squares fitting of two 3-d point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 9(5): 698–700, 1987.

[8] P. Azad, A. Ude, R. Dillmann, and G. Cheng. A full body human motion capture system using particle filtering and on-the-fly edge detection. In *Proc. of International Conference of Humanoid Robots*, pages 941–959, 2004.

[9] B. Babenko, M. Yang, and S. Belongie. Robust object tracking with online multiple instance learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 33(8):1619–1632, 2011.

[10] X. Bai, W. Liu, X. Wang, L. J. Latecki, and Z. Tu. Active skeleton for non-rigid object detection. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 1550–5499, 2009.

[11] J. Bandouch, F. Engstler, and M. Beetz. Accurate human motion capture using an ergonomics-based anthropometric human model. In *Proc. of International Conference on Articulated Motion and Deformable Objects (AMDO)*, pages 248–258, 2008.

[12] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features. *Journal of Computer Vision and Image Understanding (CVIU)*, 10(3):346–359, 2004.

[13] B. Benfold and I. Reid. Stable multi-target tracking in real-time surveillance video. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3457–3464, 2011.

[14] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua. Multiple object tracking using k-shortest paths optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 33(9):1806–1819, 2011.

[15] I. Biederman and E. E. Cooper. Evidence for complete translational and reflectional invariance in visual object priming. *Journal of Perception*, 32(12):585–593, 1991.

[16] I. Biederman and P. C. Gerhardstein. Recognizing depth-rotated objects: Evidence and conditions for three-dimensional viewpoint invariance. *Journal of Experimental Psychology: Human Perception and Performance*, 19(6):1162–1182, 1993.

[17] J. Brookshire and S. Teller. Articulated pose estimation via over-parametrization and noise projection. In *Proc. of Robotics: Science and Systems Conference (RSS)*, 2014.

[18] T. Brox, B. Rosenhahn, J. Gall, and D. Cremers. Combined region- and motion-based 3d tracking of rigid and articulated objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 32(3):402–415, 2010.

[19] D. Brščić and H. Hashimoto. Mobile robot as physical agent of intelligent space. *Journal of Computing and Information Technology*, 17(1):81—94, 2009.

[20] S. Calinon, F. Guenter, and A. Billard. On learning, representing, and generalizing a task in a humanoid robot. *Journal of Systems, Man, and Cybernetics, Part B: Cybernetics*, 37(2):286–298, 2007.

[21] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. Brief: Binary robust independent elementary features. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 778–792, 2010.

[22] A. Carballo, A. Ohya, and S. Yuta. Multiple people detection from a mobile robot using double layered laser range finders. In *Proc. of IEEE International Conference on Robotics and Automation (ICRA), Workshop on People Detection and Tracking*, 2009.

[23] Fu Chang, Chun-Jen Chen, and Chi-Jen Lu. A linear-time component-labeling algorithm using contour tracing technique. *Computer Vision and Image Understanding*, 93(2):206 – 220, 2004.

[24] L. Chen, G. Panin, and A. Knoll. Human body orientation estimation in multiview scenarios. In *Proc. of International Symposium on Visual Computing (ISVC)*, pages 499–508, 2012.

[25] L. Chen, W. Wang, and A. Knoll. Global optimal data association for multiple people tracking. In *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pages 4728–4734, 2013.

[26] G. K.M. Cheung, S. Baker, and T. Kanade. Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 77–84, 2003.

[27] C. Choi and H. I. Christensen. 3d pose estimation of daily objects using an rgb-d camera. In *Proc. of IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 3342–3349, 2012.

[28] C. Chu, O. C. Jenkins, and M. J. Mataric. Markerless kinematic model and motion capture from volume sequences. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 475–482, 2003.

[29] U. Clarenz, M. Rumpf, and A. Telea. Robust feature detection and local classification for surfaces based on moment analysis. *IEEE Transactions on Visualization and Computer Graphics*, 5(516–524), 10.

[30] D. Crandall and J. Luo. Robust color object detection using spatial-color joint probability functions. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1443–1453, 2004.

[31] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 886–893, 2005.

[32] E. de Aguiar, C. Theobalt, C. Stoll, and H.-P. Seidel. Marker-less deformable mesh tracking for human shape and motion capture. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.

[33] R. Pimentel de Figueiredo, P. Moreno, A. Bernardino, and J. Santos-Victor. Multi-object detection and pose estimation in 3d point clouds: A fast grid-based bayesian filter. In *Proc. of International Conference on Robotics and Automation (ICRA)*, pages 4250–4255, 2013.

[34] J. Deutscher, A. Blake, and I. D. Rei. Articulated body motion capture by annealed particle filtering. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 126–133, 2000.

[35] K. C. J. Dietmayer, J. Sparbert, and D. Streller. Model based object classification and object tracking in traffic scenes from range images. In *Proc. of IEEE Intelligent Vehicles Symposium*, pages 25–30, 2001.

[36] H. Eberhardt, V. Klumpp, and U. D. Hanebeck. Density trees for efficient nonlinear state estimation. In *Proc. of International Conference on Information Fusion*, pages 1–8, 2010.

[37] S. Edelman and H. H. Bülthoff. Orientation dependence in the recognition of familiar and novel view of three-dimensional objects. *Journal of Vision Research*, 12(2385–2400), 32.

[38] R. Ellis, D. A. Allport, G. W. Humphreys, and J. Collis. Varieties of object constancy. *ournal of Experimental Psychology*, 41(4):775–796, 1989.

[39] Z. Fan and B. Lu. Fast recognition of multi-view faces with feature selection. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 76–81, 2005.

[40] A. Fathi and G. Mori. Human pose estimation using motion exemplars. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 1–8, 2007.

[41] J. Fiser and I. Biederman. Size invariance in visual object priming of gray-scale images. *Journal of Perception*, 24(7):741–748, 1995.

[42] R. B. Fisher. Change detection in color images. In *Proc. of International Conference on Computer Vision (ICCV)*, 1999.

[43] E. Frontoni and P. Zingaretti. Visual feature group matching for autonomous robot localization. In *Proc. of International Conference on Image Analysis and Processing (ICIAP)*, pages 197–204, 2007.

[44] H. Fujiyoshi and A. J. Lipton. Real-time human motion analysis by image skeletonization. In *Proc. of IEEE Workshop on Application of Computer Vision (WACV)*, pages 15–21, 1998.

[45] J. Gall, B. Rosenhahn, T. Brox, and H.-P. Seide. Optimization and filtering for human motion capture. *International Journal of Computer Vision (IJCV)*, 87(1–2): 75–92, 2010.

[46] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann. Robust global registration. In *Proc. of Eurographics Symposium on Geometry Processing*, number 197, 2005.

[47] T. Gevers. Robust histogram construction from color invariants for object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 26 (1):113–118, 2004.

[48] T. Gevers and A. W. M. Smeulders. Color-based object recognition. *International Journal of Pattern Recognition (IJPR)*, 32(453–464), 1999.

[49] S. Ghosh, M. Loper, E. B. Sudderth, and M. J. Black. From deformations to parts: Motion-based segmentation of 3d objects. In *Proc. of Advances in Neural Information Processing Systems (NIPS)*, pages 2006–2014, 2012.

[50] D. F. Glas, T. Miyashita, H. Ishiguro, and N. Hagita. Laser tracking of human body motion using adaptive shape modeling. In *Proc. of IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 603–608, 2007.

[51] D. Glasner, M. Galun, S. Alpert, R. Basri, and G. Shakhnarovich. Viewpoint-aware object detection and continuous pose estimation. *International Journal of Image and Vision Computing (JIVC)*, 30(12):923–933, 2012.

[52] G. Grisetti, C. Stachniss, and W. Burgard. Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Transactions on Robotics*, 23(1):34–46, 2007.

[53] L. Guan, J. Franco, E. Boyer, and M. Pollefeys. Probabilistic 3d occupancy flow with latent silhouette cues. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1379–1386, 2010.

[54] P. Guan, A. Weiss, A. O. Balan, and M. J. Black. Estimating human shape and pose from a single image. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 1381–1388, 2009.

[55] T. Ha and C. Choi. An effective trajectory generation method for bipedal walking. *Journal of Robotics and Autonomous Systems*, 55(10):795—810, 2007.

[56] S. Hadfield and R. Bowden. Kinecting the dots: Particle based scene flow from depth sensors. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 2290–2295, 2011.

[57] P. E. Hart, N. J. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems, Science, and Cybernetics*, SSC-4(2):100–107, 1968.

[58] N. Hasler, T. Thormählen, B. Rosenhahn, and H. Seidel. Learning skeletons for shape and pose. In *Proc. of ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pages 23–30, 2010.

[59] S. Hauberg and K. S. Pedersen. Predicting articulated human motion from spatial processes. *International Journal of Computer Vision (IJCV)*, 94(3):317–334, 2011.

[60] E. Herbst, X. Ren, and D. Fox. Rgb-d flow: Dense 3-d motion estimation using color and depth. In *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pages 2276–2282, 2013.

[61] A. Hilsmann and P. Eisert. Tracking deformable surfaces with optical flow in the presence of self occlusion in monocular image sequences. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, number 1–6, 2008.

[62] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Proc. of Asian Conference on Computer Vision (ACCV)*, pages 548–562, 2012.

[63] M. Hofmann and D. M. Gavrila. Multi-view 3d human pose estimation combining single-frame recovery, temporal integration and model adaptation. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2214–2221, 2009.

[64] J. Huang and M. Yang. Estimating human pose from occluded images. In *Proc. of Asian conference on Computer Vision (ACCV)*, pages 48–60, 2009.

[65] S. Huang, L. Fu, and P. Hsiao. Region-level motion-based background modeling and subtraction using mrfs. *IEEE Transactions on Image Processing (TIP)*, 16(5): 1446–1456, 2007.

[66] X. Huang, L. Walker, and S. Birchfield. Occlusion-aware reconstruction and manipulation of 3d articulated objects. In *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pages 1365–1371, St. Paul, Minnesota, May 2012.

[67] F. Huguet and F. Devernay. A variational method for scene flow estimation from stereo sequences. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 1–7, 2007.

[68] Y. Iwashita, R. Kurazume, T. Mori, M. Saito, and T. Hasegawa. Model-based motion tracking system using distributed network cameras. In *Proc. of International Conference on Robotics and Automation (ICRA)*, pages 3020–3025, 2010.

[69] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, and A. Fitzgibbon. Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth camera. In *Proc. of ACM Symposium on User Interface Software and Technology*, pages 559–568, 2011.

[70] A. Jain and C. C. Kemp. Improving robot manipulation with haptic data from humans and robots. *International Journal of Autonomous Robots (IJAR)*, 35(2–3): 143–159, 2013.

[71] H. Jiang, T. Tian, K. He, and S. Sclaroff. Scale resilient, rotation invariant articulated object matching. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 143–150, 2012.

[72] M. Kalakrishnan, L. Righetti, P. Pastor, and S. Schaal. Learning force control policies for compliant manipulation. In *Proc. of IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 4639–4644, 2011.

[73] A. Kanezaki, T. Harada, and Y. Kuniyoshi. Partial matching of real textured 3d objects using color cubic higher-order local auto-correlation features. *Journal of the Visual Compter*, 26(10):1269–1281, 2010.

[74] A. Kanezaki, Z. Marton, D. Pangercic, T. Harada, Y. Kuniyoshi, and M. Beetz. Voxelized shape and color histograms for rgb-d. In *Proc. of IEEE International Conference on Intelligent Robots and Systems (IROS), Workshop on Active Semantic Perception and Object Search in the Real World*, 2011.

[75] J. Kato, T. Watanabe, S. Joga, Y. Liu, and H. Hase. An hmm/mrf-based stochastic framework for robust vehicle tracking. *IEEE Transactions on Intelligent Transportation Systems*, 5(3):142–154, 2004.

[76] D. Katz and O. Brock. Manipulating articulated objects with interactive perception. In *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pages 272–277, Pasadena, CA, 2008.

[77] J. Kim, C. Liu, and F. Sha. Deformable spatial pyramid matching for fast dense correspondences. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2307–2314, 2013.

[78] S. Knoop, S. Vacek, and R. Dillmann. Modeling joint constraints for an articulated 3d human body model with artificial correspondences in icp. In *Proc. of International Conference of Humanoids*, pages 74–79, 2005.

[79] S. Knoop, S. Vacek, K. Steinbach, and R. Dillmann. Sensor fusion for model based 3d tracking. In *Proc. of International Conference on Multi-sensor Fusion and Integration for Intelligent Systems*, pages 524–529, 2006.

[80] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 28(10):1568–1583, 2006.

[81] M. Kolomenkin, I. Shimshoni, and A. Tal. On edge detection on surfaces. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2767–2774, 2009.

[82] A. Kushal, C. Schmid, and J. Ponce. Flexible object models for category-level 3d object recognition. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.

[83] K. Lai, L. Bo, X. Ren, and Dieter Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *Prof. of IEEE International Conference on Robotics and Automation (ICRA)*, 2011.

[84] C.-S. Lee and A. Elgammal. Coupled visual and kinematic manifold models for tracking. *International Journal of Computer Vision (IJCV)*, 87(1–2):118–139, 2010.

[85] Dongheui Lee and C. Ott. Incremental kinesthetic teaching of motion primitives using the motion refinement tube. *Journal of Autonomous Robots*, 31(2):115–131, 2011.

[86] B. Li, M. Ayazoglu, T. Mao, and O. I. Camps. Activity recognition using dynamic subspace angles. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3193–3200, 2011.

[87] J. Liebelt, C. Schmid, and K. Schertler. Viewpoint-independent object class detection using 3d feature maps. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008.

[88] H. Ling and D. W. Jacobs. Deformation invariant image matching. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 1466–1473, 2005.

[89] Z. Liu and G. v. Wichert. Extracting semantic indoor maps from occupancy grids. *International Journal of Robotics and Autonomous Systems*, 62(5):663—674, 2014.

[90] Z. Liu, W. Wang, D. Chen, and G. v. Wichert. A coherent semantic mapping system based on parametric environment abstraction and 3d object localization. In *Proc. of European Conference on Mobile Robots (ECMR)*, pages 234–239, 2013.

[91] C. Lovato, U. Castellani, and A. Giachetti. Automatic segmentation of scanned human body using curve skeleton analysis. In *Proc. of International Conference on Computer Vision/Computer Graphics CollaborationTechniques*, pages 34–45, 2009.

[92] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 60(2):91–110, 2004.

[93] E. Lutscher, M. Lawitzky, G. Cheng, and S. Hirche. A control strategy for operating unknown constrained mechanisms. In *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pages 819–824, Anchorage, Alaska, USA, 2010.

[94] M. Matsumoto and S. Yuta. 3d laser range sensor module with roundly swinging mechanism for fast and wide view range image. In *Proc. of IEEE International Conference on Multi-sensor Fusion and Integration for Intelligent Systems*, pages 156–161, 2010.

[95] T. Matsumoto, M. Shimosaka, H. Noguchi, T. Sato, and T. Mori. Pose estimation of multiple people using contour features from multiple laser range finders. In *Proc. of IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 2190–2196, 2009.

[96] N. J. Mitra, L. J. Guibas, J. Giesen, and M. Pauly. Probabilistic fingerprints for shapes. In *Proc. of Eurographics Symposium on Geometry Processing*, pages 121–130, 2006.

[97] T. B. Moeslund and E. Granum. A survey of computer vision-based human motion capture. *International Journal of Computer Vision and Image Understanding*, 81(3):230–268, 2001.

[98] F. Moreno-Noguer, M. Salzmann, V. Lepetit, and P. Fua. Capturing 3d stretchable surfaces from single images in closed form. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1842–1849, 2009.

[99] O. M. Mozos, R. Kurazume, and T. Hasegawa. Multi-layer people detection using 2d range data. *International Journal of Social Robotics (IJSR)*, 2(1):31–40, 2010.

[100] Q. Mühlbauer, K. Kühnlenz, and M. Buss. A model-based algorithm to estimate body poses using stereo vision. In *Proc. of Robot and Human Interactive Communication (RO-MAN)*, pages 285–290, 2008.

[101] T. Naseer, L. Spinello, W. Burgard, and C. Stachniss. obust visual robot localization across seasons using network flows. In *Proc. of AAAI Conference on Artificial Intelligence (AAAI)*, pages 1–8, 2014.

[102] C. Nastar and N. Ayache. Frequency-based nonrigid motion analysis: Application to four dimensional medical images. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 18(11):1067–1079, 1996.

[103] R. Navaratnam, A. Thayananthan, P. H. S. Torr, and R. Cipolla. Hierarchical part-based human body pose estimation. In *Proc. of British Machine Vision Conference (BMVC)*, pages 479–488, 2005.

[104] A. Nüchter, K. Lingemann, J. Hertzberg, and H. Surmann. Accurate object localization in 3d laser range scans. In *Proc. of International Conference of Advanced Robotics*, pages 665–672, 2005.

[105] A. Park and K. Jung. Human pose recognition using chamfer distance in reduced background edge for human-robot interaction. In *Proc. of International Symposium on Advances in Visual Computing (ISVC)*, pages 726–735, 2010.

[106] M. Pauly, R. Keiser, and M. Gross. Multi-scale feature extraction on point-sampled surfaces. *Computer Graphics Forum*, 22(3):281–289, 2003.

[107] N. Payet and S. Todorovic. From contours to 3d object detection and pose estimation. In *Proc. of International Conference on Computer Vision (ICCV)*, number 983–990, 2011.

[108] S. Pellegrini, K. Schindler, and D. Nardi. A generalisation of the icp algorithm for articulated bodies. In *Proc. of British Machine Vision Conference (BMVC)*, pages 1–10, 2008.

[109] S. Pellegrini, L. Iocchi, O. M. Mozos, R. Kurazume, and T. Hasegawa. Multi-part people detection using 2d range data. *Journal of Social Robotics*, 2(1):31–40, 2010.

[110] J. Podolak, P. Shilane, A. Golovinskiy, S. Rusinkiewicz, and T. Funkhouser. A planar-reflective symmetry transform for 3d shapes. *ACM Transactions on Graphics, (Proceedings SIGGRAPH)*, 25(3):549–559, 2006.

[111] C. Rabe, T. Müller, A. Wedel, and U. Franke. Dense, robust, and accurate motion field estimation from stereo image sequences in real-time. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 582–595, 2010.

[112] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam. Image change detection algorithms: A systematic survey. *IEEE Transactions on Image Processing (TIP)*, 14 (3):294–307, 2005.

[113] A. Ramisa, A. Tapus, D. Aldavert, R. Toledo, and R. Mantaras. Robust vision-based robot localization using combinations of local feature region detectors. *International Journal of Autonomous Robots*, 27(4):373–385, 2009.

[114] X. Ren. Learning and matching line aspects for articulated objects. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.

[115] A. C. Romea, M. M. Torres, and S. Srinivasa. The moped framework: Object recognition and pose estimation for manipulation. *International Journal of Robotics Research (IJRR)*, 30(10):1284–1306, 2011.

[116] B. Rosenhahn, U. G. Kersting, A. W. Smith, J. K. Gurney, T. Brox, and R. Klette. A system for marker-less human motion estimation. *International Journal of Pattern Recognition (IJPR)*, 3663:109–116, 2005.

[117] E. Royer, M. Lhuillier, M. Dhome, and J. Lavest. Monocular vision for mobile robot localization and autonomous navigation. *International Journal of Computer Vision (IJCV)*, 74(3):237–260, 2007.

[118] R. B. Rusu, N. Blodow, and M. Beetz. Fast point feature histograms (fpfh) for 3d registration. In *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pages 3212–3217, 2009.

[119] R. B. Rusu, G. Bradski, R. Thibaux, and J. Hsu. Fast 3d recognition and pose using the viewpoint feature histogram. In *Proc. of IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 3467–3474, 2010.

[120] J. Ruttle, M. Manzke, and R. Dahyot. Estimating 3d scene flow from multiple 2d optical flows. In *Proc. of International Conference of Machine Vision and Image Processing Conference*, pages 1–6, 2009.

[121] K. Safronov, I. Tchouchenkov, and H. Wörn. Hierarchical iterative pattern recognition method for solving bin picking problem. In *Proc. of Robotik*, pages 3–6, 2008.

[122] S. Savarese and F. Li. 3d generic object categorization, localization and pose estimation. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 1–8, 2007.

[123] S. Schrami and A. N. Belbachir. A spatio-temporal clustering method using real-time motion analysis on event-based 3d vision. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition Workshops(CVPRW)*, pages 57–63, 2010.

[124] L. A. Schwarz, A. Mkhitaryan, D. Mateus, and N. Navab. Human skeleton tracking from depth data using geodesic distances and optical flow. *International Journal of Image and Vision Computing (JIVC)*, 30(3):217—226, 2012.

[125] A. Sellent, M. Eisemann, B. Goldlücke, D. Cremers, and M. Magnor. Motion field estimation from alternate exposure images. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 33(8):1577–1589, 2011.

[126] G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter sensitive hashing. In *Proc. of International Conference on Computer Vision (ICCV)*, volume 2, pages 750–759, 2003.

[127] A. Sharf, T. Lewiner, A. Shamir, and L. Kobbelt. On-the-fly curve-skeleton computation for 3d shapes. *Journal of Computer Graphics Forum (Proc. of Eurographics)*, 26(3):323–328, 2007.

[128] J. Shi and C. Tomasi. Good features to track. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 593–600, 1994.

[129] P. Shilane and T. Funkhauser. Selecting distinctive 3d shape descriptors for similarity retrieval. In *Proc. of IEEE Conference on Shape Modeling and Applications*, pages 18–27, 2006.

[130] Y. E. Shireen, M. E. Khaled, and H. A. Sumaya. Moving object detection in spatial domain using background removal techniques – state-of-art. *Journal of Recent Patents on Computer Science*, pages 32–54, 2008.

[131] H. Sidenbladh, M. J. Black, and D. J. Fleet. Stochastic tracking of 3d human figures using 2d image motion. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 702–718, 2000.

[132] C. Sminchisescu and B. Triggs. Kinematic jump processes for monocular 3d human tracking. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1063–6919, 2003.

[133] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis, and Machine Vision*. Number 100–102. Chapman and Hall, 1999.

[134] S. Stalder, H. Grabner, and L. Van Gool. Dynamic objectness for adaptive tracking. In *Proc. of Asian conference on Computer Vision (ACCV)*, pages 43–56, 2012.

[135] J. Starck and A. Hilton. Correspondence labelling for wide-timeframe free-form surface matching. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 1–8, 2007.

[136] C. Stoll, N. Hasler, J. Gall, H.-P. Seidel, and C. Theobalt. Fast articulated motion tracking using a sums of gaussians body model. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 951–958, 2011.

[137] J. Stückler and S. Behnke. Efficient deformable registration of multi-resolution surfel maps for object manipulation skill transfer. In *Proc. of International Conference on Robotics and Automation (ICRA)*, 2014.

[138] J. Sturm, C. Plagemann, and W. Burgard. Unsupervised body scheme learning through self-perception. In *Proc. of International Conference on Robotics and Automation (ICRA)*, pages 3328 – 3333, 2008.

[139] J. Sturm, K. Konolige, C. Stachniss, and W. Burgard. 3d pose estimation, tracking and model learning of articulated objects from dense depth video using projected texture stereo. In *Proc. of Robotics: Science and Systems Conference (In RGB-D: Advanced Reasoning with Depth Cameras Workshop, RSS In RGB-D: Advanced Reasoning with Depth Cameras Workshop, RSS)*, 2010.

[140] J. Sturm, C. Stachniss, and W. Burgard. A probabilistic framework for learning kinematic models of articulated objects. *Journal of Artificial Intelligence Research*, 41(2):477–526, 2011.

[141] M. Sun and S. Savarese. Articulated part-based model for joint object detection and pose estimation. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 723–730, 2011.

[142] S. Sural, G. Qian, and S. Pramanik. A histogram with perceptually smooth color transition for image retrieval. In *Proc. of International Conference on Computer Vision, Pattern Recognition and Image Processing*, pages 664–667, 2002.

[143] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 30(6):1068–1080, 2008.

[144] A. Tagliasacchi, H. Zhang, and D. Cohen-Or. Curve skeleton extraction from incomplete point cloud. *ACM Transactions on Graph*, 28(3):71, 2009.

[145] H. Tamura, T. Sasaki, H. Hashimoto, and F. Inoue. Position measurement system for cylindrical objects using laser range finder. In *Proc. of SICE Annual Conference*, pages 291–296, 2010.

[146] J. Tang, S. Miller, A. Singh, and P. Abbeel. A textured object recognition pipeline for color and depth image data. In *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pages 3467–3474, 2012.

[147] A. Tevs, M. Bokeloh, M. Wand, A. Schilling, and H.-P. Seidel. Isometric registration of ambiguous and partial data. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1185–1192, 2009.

[148] A. Thomas, V. Ferrari, B. Leibe, T. Tuytelaars, B. Schiele, and L. V. Gool. Towards multi-view object class detection. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1589–1596, 2006.

[149] S. Thrun and B. Wegbreit. Shape from symmetry. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 1824–1831, 2005.

[150] F. Tombari, S. Salti, and L. D. Stefano. Unique signatures of histograms for local surface description. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 356–369, 2010.

[151] F. Tombari, S. Salti, and L. D. Stefano. A combined texture-shape descriptor for enhanced 3d feature matching. In *Proc. of IEEE International Conference on Image Processing (ICIP)*, pages 809–812, 2011.

[152] Z. Tu, X. Chen, A. L. Yuille, and S. Zhu. Image parsing: Unifying segmentation, detection, and recognition. *International Journal of Computer Vision (IJCV)*, 63(2): 113–140, 2005.

[153] T. Tung and T. Matsuyama. Dynamic surface matching by geodesic mapping for 3d animation transfer. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, number 1402–1409, 2010.

[154] R. Urtasun, D. J. Fleet, and P. Fu. 3d people tracking with gaussian process dynamical models. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 238–245, 2006.

[155] A. Vadivel, A. K. Majumdar, and S. Sural. Perceptually smooth histogram generation from the hsv color space for content based image retrieval. In *Proc. of Advances in Pattern Recognition*, pages 248–251, 2003.

[156] L. Valgaerts, A. Bruhn, H. Zimmer, J. Weickert, C. Stoll, and C. Theobalt. Joint estimation of motion, structure and geometry from stereo sequences. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 568–581, 2010.

[157] S. Vedula, P. Rander, R. Collins, and T. Kanade. Three-dimensional scene flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):475–480, 2005.

[158] C. J. Veenman, M. J. T. Reinders, and E. Backer. Resolving motion correspondence for densely moving points. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 23:54–72, 2001.

[159] C. Wang, M. M. Bronstein, A. M. Bronstein, and N. Paragios. Discrete minimum distortion correspondence problems for non-rigid shape matching. In *Proc. of International Conference on Scale Space and Variational Methods in Computer Vision*, pages 580–591, 2012.

[160] W. Wang, D. Brščić, Z. He, S. Hirche, and K. Kühnlenz. Real-time human body motion estimation based on multi-layer laser scans. In *Proc. of IEEE International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, pages 297–302, 2011.

[161] W. Wang, L. Chen, D. Chen, S. Li, and K. Kühnlenz. Fast object recognition and 6d pose estimation using viewpoint oriented color-shape histogram. In *Proc. of IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2013.

[162] W. Wang, L. Chen, Z. Liu, K. Kühnlenz, and D. Burschka. Textured/textureless object recognition and pose estimation using rgb-d image. *Journal of Real-Time Image Processing (JRTIP)*, 2013.

[163] W. Wang, V. Koropouli, Dongheui Lee, and K. Kühnlenz. Articulated object modeling based on visual and haptic observations. In *Proc. of International Conference on Computer Vision Theory and Applications (VISAPP)*, pages 253–259, 2013.

[164] A. D. Ward and G. Hamarneh. The groupwise medial axis transform for fuzzy skeletonization and pruning. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 32(6):1084 – 1096, 2010.

[165] A. Wedel, T. Brox, T. Vaudrey, C. Rabe, U. Franke, and D. Cremers. Stereoscopic scene flow computation for 3d motion understanding. *International Journal of Computer Vision (IJCV)*, 95(1):29–51, 2011.

[166] L. Weiss and M. Ray. Recognizing articulated objects using a region-based invariant transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 27(10):1660–1665, 2005.

[167] W. Wohlkinger and M. Vincze. Ensemble of shape functions for 3d object classification. In *Proc. of IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 2987–2992, 2011.

[168] K. Xu, H. Zhang, A. Tagliasacchi, L. Liu, G. Li, M. Meng, and Y. Xiong. Partial intrinsic reflectional symmetry of 3d shapes. *ACM Transactions on Graphics, (Proceedings SIGGRAPH Asia)*, 28(5):138:1–138:10, 2009.

[169] M. Ye, X. Wang, R. Yang, L. Ren, and M. Pollefeys. Accurate 3d pose estimation from a single depth image. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 731—738, 2011.

[170] Z. Yin and R. Collins. Belief propagation in a 3d spatio-temporal mrf for moving object detection. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.

[171] A. Zaharescu, E.Boyer, K.Varanasi, and R.Horaud. Surface feature detection and description with applications to mesh matching. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 373–380, 2009.

[172] Y. Zeng, C. Wang, Y. Wang, X. Gu, D. Samaras, and N. Paragios. Intrinsic dense 3d surface tracking. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1225–1232, 2011.

[173] L. Zhang, J. Sturm, D. Cremers, and D. Lee. Real-time human motion tracking using multiple depth cameras. In *Proc. of International Conference on Intelligent Robot Systems (IROS)*, pages 2389–2395, 2012.

[174] Q. Zhang, X. Song, X. Shao, R. Shibasaki, and H. Zhao. Unsupervised skeleton extraction and motion capture from 3d deformable matching. *Neurocomputing*, 100:170–182, 2013.

[175] X. Zhang, D. Chen, Z. Yuan, and N. Zheng. Dense scene flow based on depth and multi-channel bilateral filter. In *Proc. of Asian Conference on Computer Vision (ACCV)*, pages 140–151, 2012.

[176] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision (IJCV)*, 13(2):119–152, 1994.

[177] X. Zhao and Y. Liu. Generative estimation of 3d human pose using shape contexts matching. In *Proc. of Asian conference on Computer Vision (ACCV)*, pages 419–429, 2007.

[178] Q. Zheng, A. Sharf, A. Tagliasacchi, B. Chen, H. Zhang, A. Sheffer, and D. Cohen-Or. Consensus skeleton for non-rigid space-time registration. *Journal of Computer Graphcis Forum (Special Issue of Eurographics)*, 29(2):635–644, 2010.

[179] F. Zhou, F. De la Torre, and J. K. Hodgins. Hierarchical aligned cluster analysis for temporal clustering of human motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 35(3):582–596, 2013.

[180] R. Zhu and Z. Zhou. A real-time articulated human motion tracking using tri-axis inertial/magnetic sensors package. *IEEE Transaction on Neural Systems and Rehabilitation Engineering*, 12(2):295–302, 2004.

[181] J. Ziegler, K. Nickel, and R. Stiefelhage. Tracking of the articulated upper body on multi-view stereo image sequences. In *Proc. of IEEE Annual Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 774–781, 2006.

[182] A. Zweng and M. Kampel. Unexpected human behavior recognition in image sequences using multiple features. In *Proc. of International Conference on Pattern Recognition (ICPR)*, pages 368–371, 2010.