TECHNISCHE UNIVERSITÄT MÜNCHEN
Lehrstuhl für Echtzeitsysteme und Robotik

Efficient 3D Human Motion Perception System with Un-supervision, Randomization and Discrimination

Guang Chen

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

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender:	Univ.-Prof. Dr. Tobias Nipkow, Ph.D.
Prüfer der Dissertation:	1. Univ.-Prof. Dr.-Ing. habil. Alois Knoll
	2. Prof. Dr. Hong Qiao, Chinese Academy of Sciences/China

Die Dissertation wurde am 28.12.2016 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 25.04.2017 angenommen.

Abstract

Building a 3D human motion perception system in an efficient way remains non-trivial. The system must not only automatically interpret new modality data without too much human intervention, but it also needs to yield high performance results in the absence of professional knowledge of the data. The system must not only explore meaningful information from multiple modalities, but it also need to cope with the rich and dense representations from then. To achieve the efficiency, advanced machine learning techniques have been developed that are usually able to generate and mine useful knowledge from empirical example data.

There are many existing approaches for 2D human motion perception system by utilizing machine learning techniques. However, their input modality data differ in many aspects. 2D modality data have its intrinsic limitations, such as sensitive to illumination changes and lack of 3D structural information. Therefore, directly extending the solutions of 2D human motion perception system is inefficient for the new modality.

This work presents two frameworks for the development of 3D human motion perception systems: autonomously feature learning with un-supervision and efficiently system developing with randomization and discrimination. Un-supervised feature learning framework serves as a basis for low-level 3D video representation and is able to leverage the plethora of the unlabeled data and adapt easily to new modality data. Efficient system developing framework mines the most useful feature and model the 3D human motion in a joint way with randomization and discrimination. It is able to discover semantically meaningful knowledge over the rich representations that closely matches the intuition of human vision system.

The major contribution of this thesis is the presentation of three advanced machine learning techniques, *un-supervision, discrimination and randomization*, to guarantee the efficiency when developing 3D human motion perception system. With *un-*

supervision, input modality data allows to be processed automatically without knowing the intrinsic properties of the data. With *discrimination*, meaningful knowledge is automatically mined from input data. With *randomization*, the algorithm is able to cope with rich and dense representation in an efficient way.

Two applications of 3D human action and gesture recognition system are developed to demonstrate the practicability of the approach. *Un-supervision, discrimination and randomization* are implemented and realized in this context. Applications demonstrate the advantages of developed advanced machine learning techniques, achieve comparable performance with state-of-the-art method, and validate the leading principle of this thesis, the *efficiency* of 3D human motion perception system.

Zusammenfassung

Ein Wahrnehmungssystem für menschliche 3D Bewegungen effizient zu entwickeln ist weiterhin anspruchsvoll. Das System muss nicht nur automatisch neue Daten unterschiedlicher Art ohne zu viel menschlichen Unterstützung interpretieren, sondern es muss auch ohne Expertenwissen über die Daten noch Ergebnisse mit hoher Zuverlässigkeit liefern. Das System muss nicht nur bedeutsame Informationen in unterschiedlichartigen Inputdaten finden, es muss auch in der Lage sein, mit informationsreichen und dichten Darstellungen solcher Inputdaten umzugehen. Um diese Effizienz zu erreichen, wurden fortschrittliche Machine-Learning-Ansätze entwickelt, die in der Lage sind, nützliches Wissen aus empirischen Beispieldaten zu erzeugen und zu extrahieren.

Es gibt zahlreiche existierende Ansätze für Wahrnehmungssysteme menschlicher Bewegung in 2D durch die Nutzung von Machine-Learning-Techniken. Jedoch sind die zahlreichen Arten von Inputdaten in vielen Aspekten verschieden. 2D-Daten haben inhärente Einschränkungen wie Empfindlichkeit gegen Belichtungsänderungen und die Abwesenheit von strukturellen 3D-Informationen. Deswegen ist es ineffizient, die Lösungen für 2D-Wahrnehmungssysteme menschlicher Bewegung für neue Inputdaten direkt zu erweitern.

Diese Arbeit stellt zwei Frameworks zur Entwicklung von 3D-Wahrnehmungssystemen menschlicher Bewegungen vor: autonomes Lernen von Features ohne Überwachung und effiziente Systementwicklung mit Randomisierung und Diskriminierung. Ein "Unsupervised Feature Learning Framework"dient als Basis für eine low-level 3D-Video-Darstellung und ist in der Lage, die Vielzahl von Daten ohne Labels auszunutzen und sich leicht an neue, verschiedenartige Daten anzupassen. Ein effizientes Systementwicklungsframework extrahiert die nützlichsten Features und modelliert die menschliche 3D-Bewegung zusammen mit Randomisierung und Diskriminierung. Es ist in der Lage, bedeutungsreiches Wissen semantisch aus informationsreichen

Darstellungen zu gewinnen, die der Intuition des menschlichen visuellen Systems sehr nahekommen.

Der größte Beitrag dieser Arbeit ist die Vorstellung dreier fortschrittlicher Machine-Learning-Techniken, *Nicht-Supervision, Diskriminierung und Randomisierung*, die drei Schlüsselkomponenten sind, um die Effizienz bei der Entwicklung von Wahrnehmungssystemen menschlicher Bewegung in 3D zu gewährleisten. Mit *Nicht-Supervision* ist es möglich, verschiedenartige Inputdaten automatisch zu verarbeiten ohne intrinsische Eigenschaften dieser Daten zu kennen. Mit Hilfe von *Diskriminierung* wird automatisch bedeutungsvolles Wissen aus den Inputdaten gewonnen. Mittels *Randomisierung* ist der Algorithmus in der Lage mit informationsreichen und dichten Representationen effizient umzugehen.

Zwei Anwendungen des Erkennungssystems für menschliche 3D-Handlungen und Gesten wurden entwickelt, um die Anwendbarkeit des Ansatzes unter Beweis zu stellen. *Nicht-Supervision, Diskriminierung und Randomisierung* wurden in diesem Kontext implementiert und umgesetzt. Anwendungen demonstrieren die Vorteile der entwickelten fortschrittlichen Machine-Learning-Techniken. Diese erzielen Ergebnisse, die vergleichbar mit State-of-the-Art Methoden sind und validieren das Leitprinzip dieser Arbeit, die *Effizienz* des Wahrnehmungssystems für menschliche Bewegungen.

Acknowledgements

First of all, I would like to express my deepest gratitude to my adviser Prof. Dr. Alois Knoll, for his guidance, support, and encouragement. I am very thankful to Prof. Dr. Hong Qiao for investing time in reviewing my thesis and acting as a second supervisor. I am very grateful that my master thesis advisor, Prof. Zhihua Zhong recommends and supports me to pursue a PhD degree in Germany.

Many thanks go to the Data Fusion Group at fortiss and Neurorobotic Group at the TU München for the valuable discussions and the pleasant atmosphere. In particularly, I would like to thank Dr. Christian Buckl, Dr. Daniel Clarke and Dr. Florian Röhrbein for their guidance and support. I would like to thank the data fusion group/FB2 members at fortiss Dr. Feihu Zhang, Gereon Hinz, Hauke Stähle, Dr. Chao Chen, Dr. Andre Gaschler, Axel von Arnim, and Dr. Manuel Giuliani. I would like to thank the neurorobotic group/embedded systems and robotics members Florian Walter, Kenny Sharma, Mahmoud Akl, Ievgen Smielik, Alexander Kuhn, Zhenshan Bing, Biao Hu, Xuebing Wang, Long Chen, Mingchuan Zhou, Caicai Xia and Dr. Kai Huang. I would also like to thank all the other members at the chair, especially Amy Bücherl, and Ute Lomp, for their very kind help and support.

I want to express special thanks to all my colleagues and friends for proof-reading my thesis and for their good comments that helped me improving it.

Finally, I would like to thank my wife Yanping for her continuous support, patient, love and encouragement. My son Xiaoshitou cheered me up after long hours in the lab. I am thankful to my sister Jiao Chen, my parents Maijun Chen and Ruihua Jiao for all their love and efforts in educating me.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Thesis Motivation

Machine perception of human motion plays a crucial role in many artificial intelligence systems: health monitoring system shall recognize activity changes of patients, sign language translation system supports communication between deaf and hearing persons and human vehicle interaction system can perform secondary driving tasks by recognizing driver's hand gestures. For all these tasks the ability to autonomously percept the human motions (e.g., actions, gestures, expressions) by a machine under different environments is the key requirement. This very basic ability is one of the most important social skills in our everyday life. Human, being highly social creatures, easily possess these perceptual skills: to perceive what others are doing and to infer from actions, gesture and expressions what others may be intending to do, but still poses a lot of questions and open problems for intelligent machines.

The scientific field of computer vision community tackles these issues and has developed many vision-based techniques and methodologies towards solutions for the past few decades. The central goal of computer vision scientists is to enable the machine to duplicate the human vision's ability to automatically understand the surroundings using vision sensors. The first study of human motion was provided by Johansson [1]'s pioneering work in the early 1970s, and this work has been studied for the recognition of human activities in depth by Webb and Aggarwal [2] in the early 1980s. Since 1999, human activity recognition was in its infancy, as Aggarwal and Cai [3] pointed out. It was at a stage where more and more experimental systems were deployed at airports and other public places. A significant amount of progress on human activity recognition achieved by computer vision researchers, but it is still far from being an

off-the-shelf technology due to the limited amount of the video data and computing limitations of smart sensors and devices.

In the 2000s, along with the technological advances of computers in general, video data become more and more accessible and play an increasingly important role in everyday life. Even commonly used consumer hardware, such as notebooks, mobile phones, and digital photo cameras, allow to create videos. For example, the amount of video uploaded to YouTube every minutes increased from six hours in 2007, 20 hours in 2009 to 500 hours in 2015[1]. During this stage various approaches have been proposed addressing on recognizing actions and activities from 2D video data taken by visible light cameras. However, despite the increasing importance and amount of video data, the possibilities to analyze it in an automated fashion are rather limited. The abilities of the state-of-the-art computer vision system are far behind the capabilities of human vision, and face the great challenge on processing magnanimity data. One major issue with 2D videos is that capturing articulated human motion from visible-light cameras results in a considerable loss of information (e.g., lack of 3D structural information of the environment). This confines the performance of 2D video-based human motion perception system. Thus, at this stage recognizing human actions from 2D video data is still a challenge task. For instance, searching video in large-scale video databases is only possible with manual annotation information. Popular Web search engines mainly rely on textual information, such as tags or text descriptions, in order to retrieve highly relevant videos. Another example is surveillance application. According to the report[2], the city of London has installed about 1 million CCTV cameras at the cost of approximately 200 million British pounds. However, in 2008, surveillance cameras helped to solve only one crime per 1000 cameras. It has been pointed out that CCTV leads to massive expense and minimum effectiveness.

After the recent release of low-cost 3D sensing devices in 2010, people consider the task of human motion perception in 3D video data. In fact, research on 3D video data emerged in the 1990s [4]. People obtained 3D video data by using marker-based motion capture systems, stereo cameras or range sensors. However, these systems have several constraints which limit the spread of applications in human motion perception. For instance, reconstructing depth from stereo cameras requires expensive computations, and typically introduces substantial, undesirable artifacts. In contrast, depth cameras use structured light to generate real-time 3D depth video data rather reliably. The advent of depth cameras at relative inexpensive costs

[1] http://tubularinsights.com
[2] www.telegraph.co.uk/

and smaller sizes gives researchers easy access to the 3D video data at a higher frame rate resolution, leading to the rapid growth of research interests on 3D human motion perception. One obvious example is that the number of popular 2D action datasets developed by computer vision researchers before 2012 is 28, comparing to 30 3D action datasets after 2010. This thesis will also focus on human motion perception from 3D video data taken by the depth cameras.

During the past decades, machine perception of 2D human motion received significant attention of human computing task and achieved great success in the development of the techniques and methodologies. Researchers in computer vision attempt to tackle 3D human motion perception by using extensions of traditional 2D-based methods. However, we argue that these approaches are inefficient in 3D human motion perception. Although 2D action recognition has several inherent connections with 3D action recognition, they differ in many points. First, 2D video data has intrinsic limitations (e.g., it is sensitive to illumination changes). Second, single 2D sensor (the visible light camera) can not provide 3D structural information of the environment, which offers discerning information to recover postures and recognize human actions. Third, 3D video data reflect pure geometry and shape cues and provide depth information of the environment. Fourth, 3D sensors (depth sensors) have facilitated a human motion capturing technique [5] that outputs the 3D joint positions of the human skeleton. The above properties can be regarded as the advantages of the development of the 3D human motion perception systems. Of course as a newly emerging thing it has its own problems. First, the 3D video data provided by the depth cameras has some technical limitations such as spatial and temporal resolutions. Second, the 3D video data are significantly noisy and the 3D positions of the tracked joints may be wrong if serious occlusions occur. This increase the intra-class variations in 3D human actions. Third, depth cameras provide the opportunity to get a rich collections of features from multiple modalities. However, not all the features offer good discrimination among the 3D human motion perception. Also rich representations make the computational requirements more expensive. With these advantages and disadvantages in mind, in 3D human motion perception there are some challenges to master, which can be summaries as follows and be addressed in this thesis:

- *Advanced sensory technology.* The past five years have witnessed great progress in depth camera technology, which brings huge opportunities for human motion perception field. The depth video are very different from the conventional 2D video data. Novel visual representations and machine learning methods need to be developed in order to fully exploit depth sensors for human motion perceptions. New scientific tasks have to be solved

to enable next generation applications for robotics, surveillance, security, and human-computer interfaces.

- *Multi modalities and massive data.* The depth cameras provide multi modalities, such as skeleton joint data, depth data, gray-scale data, surface normal data. Which modalities and which part of human skeleton should be combined for realization of robust and accurate human motion perception? Too much information from different channels seems to be confusing for human. Does this pertain in the machines which are trained by human? In addition, the popularization of novel 3D sensing techniques leads to richer 3D human motion dataset. It is however important to note that the ground truth provided in datasets is limited due to tediously manual annotations.

- *Domain-dependent and modality-dependent problem.* One drawback of state-of-the-art human motion perception systems is that they are highly domain-dependent and modality-dependent. This issue is addressed by Wang et al. [6]. They found that there is no universally best hand-engineering feature for different dataset. Additionally, it is difficult and time-consuming to extend the existing hand-engineering feature to a new modality. An ideal solution should be fully data-driven and can learn sophisticated representations that capture statistical patterns and relationship in the dataset.

- *Redundant knowledge.* In many computer vision tasks, dense feature representations are often used to capture enough information from high-dimensional visual data. This principle can also be employed in 3D human motion perception. However, rich representations always introduce significant redundancy among the feature space, and are not discriminative for distinguishing different human motions. Therefore, some innovative data mining algorithms need to be developed to cope with the new challenges from 3D human motion perception.

The remainder of this chapter is composed as follows: the goals of this thesis are summarized in Section 1.2. Section 1.3 highlights the main contributions of this thesis. Section 1.4 gives the structure of this thesis. Throughout this thesis we will use a set of terms and abbreviations, which are explained in this section. In addition, parts of the material discussed in this thesis are published and presented at peer-reviewed conferences and journals, which will also be listed in this section.

1.2 Thesis Goals

After motivating the topic of this thesis in Section 1.1, the main goals are summarized in this section. As already stated in the title, the leading principle of this thesis is to guarantee the efficiency when developing 3D human motion perception system. Regarding to the challenges of 3D human motion perception, we follow a main thread that this thesis will not only bring the issue of perceiving human motions into 3D space, but giving professional and technical insights into how to revisit and build the 3D human motion perception system facilitating new characteristics and new challenges in an efficient way .

This thesis is to come up with a two-stage process by consisting of a 3D video representation phase and a system developing phase. Both phases take the efficiency as primary consideration. We view our leading principle, the efficiency, from three sides: an algorithm gets more efficient if it processes data automatically without human intervention while giving the high performance results, an algorithm gives better results within an optimized combination of multiple input modalities, and an algorithm achieves meaningful results by discovering the useful knowledge from a richer representation. Efficiency should be realized in different forms by developing advanced machine learning techniques. Advanced machine learning techniques refer to the process of generating and mining knowledge from empirical example data, collected by observing a process or system of interest. Here, advanced machine learning techniques will be mainly explored to cope with the noisy and ambiguous data provided by the depth cameras. In order to guarantee the efficiency, we expect that the developed advanced machine learning techniques can be used to learn the feature representation automatically from unlabeled RGBD video data. This will leverage the plethora of the unlabeled data and adapt easily to new modality data, such as depth video data. We expect that these techniques can be helpful to discover meaningful knowledge over the rich collections of spatial-temporal features, and could learn a more robust, effective and discriminative feature representation of the 3D human motions. All the techniques should be integrated into the 3D human motion perception system developed in this thesis. The system should demonstrate the advantages of the developed advanced machine learning techniques as well as achieve comparable performance with state-of-the art method although the accuracy of the system is not our primary goal in this thesis.

1.3 Thesis Contributions

The result of this thesis is the development of advanced machine learning techniques for 3D human motion perception. The main contributions of this thesis are summarized below:

- The major contribution of this thesis is the presentation of three advanced machine learning techniques, *un-supervision, discrimination and randomization*, that are three key components to guarantee the efficiency when developing 3D human motion perception system. Building a 3D human motion perception system in an efficient way is a complex task. The system must not only automatically interpret new modality data without too much human intervention, but it also need to give the high performance results in the absence of professional knowledge of the data. The system must not only discover meaningfully information from multiple input modalities, but it also need to cope with the rich and dense representations from multiple input modalities. With *un-supervision*, input modality data allows to be processed automatically without knowing the intrinsic properties of the data. With *discrimination*, meaningful knowledge is automatically mined from input data. With *randomization*, the algorithm is able to cope with rich and dense representation in an efficient way.

- We extensively explore the bio-inspired deep learning and feature learning methods for 3D video data analysis. Previous work on 3D human motion perception has focused on using hand-designed features, either from 3D videos or 2D video. We are the first attempt to extend the independent analysis algorithm to learn spatio-temporal features from the raw signal of multi-modality video data (e.g., pixels of depth video data, surface normal video data, gradient magnitude video data), and further use the ISA algorithm for the depth video data and human skeleton data together. The benefits include: we provide an efficient and powerful spatio-temporal feature representation framework for the 3D video data. Our approach is rather generic and unsupervised, and may therefore be adapted to a wider range of problems with unlabeled sensor data.

- We present a novel ensemble learning approach for 3D human action recognition, by combining unsupervised feature learning and discriminative feature mining algorithms. The basic idea of the approach is that a certain action classes are usually only associated with a subset of kinematic joints of the articulated human body. We therefore formulate the 3D action recognition task as a multiple kernel learning problem. Actions are then

represented as a linear combination of joints, where each joint associated with a weight. The benefits include: Our weighted-joint model is robust to the errors in the features, and it can better characterize the intra-class variations. We find that a small subset of joints (1-6 joints) is sufficient to perform action recognition if action classes are targeted. This observation is important for making on-line decisions and the improvements in efficiency in the action recognition task.

- We present a novel data mining algorithm for sub-ordinate human motion perception tasks, named *multimodal gesture detection and recognition with randonmization and discrimination*, which discover video blocks with arbitrary shapes, sizes or spatio-temporal localtions that carry discriminative video statistics. This approach utilizes the discriminative decision trees under the random forest framework. Unlike traditional decision trees, this approach uses strong classifiers (e.g.,support vector machines) at each node to effectively mine a very dense sampling spatio-temporal space. The randomization is used to explore the dense sampling spatio-temporal space efficiently. The benefits include: We built a multi-modality gesture detection and recognition system, which can serve as a standard framework for human motion perception task. The evaluation is performed on the largest 3D human gesture dataset in the literature: the Chalearn Gesture dataset. The result shows that the system is to able to identify semantically meaningful spatio-temporal contents that closely match human intuition. The system is generic and can even directly apply to 2D human motion perception or other sensor based human motion perception.

1.4 Thesis Structure, Terms and Abbreviations

1.4.1 Thesis Structure

This thesis is structured as in Fig. 1.1. There are six chapters. At the beginning, Chapter 1 has already summarized the goals and highlighted the main contributions of this thesis.

Following the introduction chapter of this thesis, Chapter 2 provides an in-depth study of the recent advances in 3D human motion perception. As human motion perception constitutes an active research field due to its various applications in the field of robotics, automotive industry, video surveillance and human-machine interaction, there is an wide range of emerging applications based on 3D human motion perceptions. We investigate the state-of-the-art approaches in this field to facilitate the comparison of different methods and give insight into the

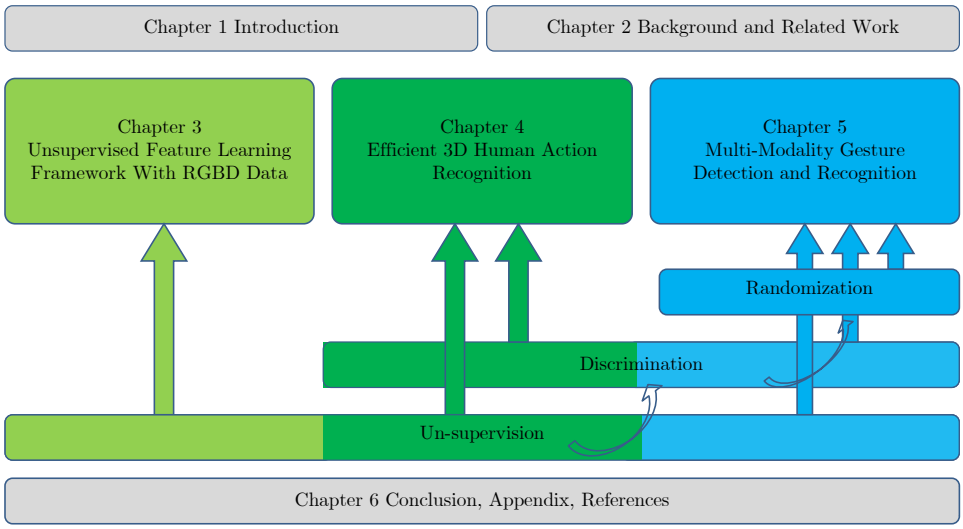


Figure 1.1: The structure of this thesis. There are six chapters. The outline also shows the leading principle of this PhD thesis: efficient 3D human motion perception through machine learning techniques *un-supervision*, *discrimination* and *randomization*, and the relationship between these three techniques and different chapters.

abilities of the different methods. We address the limitations of the state-of-the-art and point out the challenges and promising directions of future research.

In Chapter 3, we present our unsupervised feature representation framework with RGBD video and skeleton data. We will firstly discuss the state-of-art techniques employed for 3D video representations from depth sensors. We divide state-of-the-art 3D video representation methods into three categories: extended 2D-based representations, depth-based representations, skeleton-based representations. Different types of approaches address the typical characteristics for the corresponding modalities (e.g depth video data, skeleton data). We keep the advantages and disadvantages of those approaches in mind, and propose our approach for automatically learning 3D spatial-temperal feature from unlabeled RGBD video data and 3.D spatial-temperal feature from unlabeled RGBD video and skeleton data in an unsupervised way. As a general framework, it can be used by any discriminative and generative classifiers in human motion perception tasks.

Chapter 4 presents an efficient way for 3D human action recognition by combining unsupervised feature learning with discriminative feature mining. Unsupervised feature learning allows us to extract spatio-temporal features from unlabeled video ddata. With this, we can avoid

the cumbersome process of manually designed feature extraction. We propose an ensemble approach using a discriminative learning algorithm, where each base learner is a discriminative multi-kernel-learning classifier. Our evaluation includes a comparison to state-of-the-art methods on the MSRAction 3D dataset, where our method, abbreviated EnMkl, outperforms earlier methods. Furthermore, we analyze the efficiency of our approach in a 3D human action recognition system.

Chapter 5 describes a novel data mining algorithm for fine-grained human motion perception tasks. The main idea is to identify semantically meaningful spatio-temporal blocks that closely match human intuition. A multi-modality gesture detection and recognition system is built. The leading three concepts of this thesis, *un-supervision, randomization and discrimination* are validated in the chapter. The proposed system participated in the 2014 Chalearn Looking at People Challenge, which provides the world's largest 3D human gesture dataset. Our approach achieves good classification performance in this machine learning challenge. Furthermore, the system is generic and can serve as a standard framework for both 2D and 3D human motion perception.

Finally, Chapter 6 concludes this thesis with a summary of the main results of this thesis. It also gives an outlook on future research directions of 3D human motion perception.

1.4.2 Terms and Abbreviations

Throughout this thesis we will use a set of terms, which are explained in this section.

- *Modalities.* We call each source that provides information about the surrounding environment by sensors *modalities.* A modality can be a category of input data for human motion perception task. For instance, the RGB camera can provide the RGB video data and the gray-scale video data. The depth camera can provide the depth video data, the gray-scale video data, and the skeleton joint data.

- *2D video data and 3D video data.* We call the video data provided by the RGB camera as *2D video data*, and the video data provided by the depth camera as *3D video data.*

- *Human motion, human action and human gesture.* We focus on the human motion that involves both full-body movements and partial-body movements. We call the full-body movements *human actions.*, and the partial-body movements *human gestures.*

Table 1.1: Abbreviations used in this thesis.

SVM	Support Vector Machine
RF	Random Forest
SPM	Spatial Pyramid Matching
NN	Nearest Neighbor Classifiers
kNN	k-Nearest Neighbor Classifiers
BOW	Bag of Words
DTM	Dynamic Time Warping
ROI	Region of Interest
MRF	Markov Random Field
HMM	Hidden Markov Model
HOG	Histogram of Gradient
STIP	Space-time Interest Points
RBM	Restricted Boltzmann Machine
ISA	Independent Subspace Analysis
ICA	Independent Component Analysis
PCA	Principal Component Analysis
VQ	Vector Quantization
MKL	Multi Kernel Learning

- *3D Human motion perception.* We consider the task of detecting and recognizing 3D video data containing human motion with action or gesture classes. An automated detection and recognition of human actions or gestures is called *human motion perception.*

The abbreviations that we used in the text are list in Table 1.1:

1.4.3 Publications

Parts of the material discussed in this thesis were published and presented at peer-reviewed conferences and journals.

- Guang Chen, Daniel Clarke, Manuel Giuliani, Andre Gaschler, and Alois Knoll. Combining unsupervised learning and discrimination for 3d action recognition. In Signal Processing, Elsevier, 2015

- Guang Chen, Daniel Clarke, David Weikersdorfer, Manuel Giuliani, Andre Gaschler, and Alois Knoll. Multi-modality gesture detection and recognition with un-supervision, ran-

domization and discrimination. In ChaLearn Looking at People Workshop, European Conference on Computer Vision (ECCV 2014)

- Guang Chen, Manuel Giuliani, Daniel Clarke, Andre Gaschler, and Alois Knoll. Action recognition using ensemble weighted multi-instance learning. In IEEE International Conference on Robotics and Automation (ICRA 2014)

- Guang Chen, Feihu Zhang, Daniel Clarke, and Alois Knoll. Learning weighted joint-based features for action recognition using depth camera. In The 9th International Conference on Computer Vision Theory and Applications(VISAPP 2014)

- Guang Chen, Feihu Zhang, Manuel Giuliani, Christian Buckl, and Alois Knoll. Unsupervised learning spatio-temporal features for human activity recognition from rgb-d video data. In Proceedings of the International Conference on Social Robotics (ICSR 2013)

Chapter 2

Background and Related Work

In this chapter, we introduce the background of this thesis. In Section 2.1, we briefly compare 2D human motion perception to 3D human motion perception. We simply discuss the differences between each other. After that, In Section 2.2 we investigate the state-of-the-art approaches in the field of 3D human motion perception to facilitate the comparison of different methods and give insight into the development of the 3D human motion perception system in this thesis.

2.1 Human Motion Perception: from 2D to 3D

During the past decades, machine recognition of 2D human actions received significant attention of human computing tasks [7], as human actions are the most natural means for humans to regulate interactions with the environment and other subjects. Research has mainly focused on recognizing actions from 2D video data taken by visible light cameras (e.g., [8, 9, 10]). Numerous 2D vision datasets (e.g., [11, 12, 13, 14, 15]) are created to promote the development of human action recognition. There are a bunch of approached developed in the area of 2D action recognition [7, 9, 16, 17, 18, 19]. The interest in the human motion perception is further promoted by the increasing popularity of novel 3D sensing devices. In addition, new techniques allow the easy and affordable acquisation of 3D video data. For example, many 3D sensors (depth sensors) now facilitate a human motion capturing technique [5] that outputs the 3D joint positions of the human skeleton.Moreover, more and more 3D datasets dedicated to human motion perception have been created. As a result, many researchers have transfered their attentions from 2D human motion perception to 3D human motion perception. Although 2D action recognition has several inherent connections with 3D action recognition, they differ in many points. First, 2D video data has intrinsic limitations (e.g., it is sensitive to illumination

changes). Second, single 2D sensor (the visible light camera) can not provide 3D structural information of the environment, which offers discerning information to recover postures and recognize human actions. Alternatively, 3D video data reflect pure geometry and shape cues and provide depth information of the environment. In order to exploit the benefits of new multiple modalities video data, new methodologies are developed which differ greatly from the methodologies of 2D human motion perception. The remaining of this Chapter will provide a comprehensive study of state-of-the-art 3D human motion analysis methodologies. We focus on recent work in machine recognition of human motions based on 3D datasets (3D video data recorded by depth sensors), which is also the research topic in the following chapters of this thesis.

2.2 3D Human Motion Perception

In this section, we discuss state-of-the-art approaches for 3D human motion perception. In section 2.2.1, we firstly introduce data mining approaches used in 3D action recognition. Data mining as a common step before the actual classification is springing up in recently developed approaches for 3D action recognition. It is used to cope with the new challenges (e.g., noisy data, rich collection of features) from the new 3D modalities. In section 2.2.2, we discuss approaches that classify 3D video representations into action classes without explicitly modelling temporal variations. In section 2.2.3, we discuss temporal state-space approaches that model such variations of 3D actions. In Section 2.2.4, we describe general approaches for 3D action detection (e.g., detecting past actions) and 3D action anticipation (e.g., anticipating future action).

2.2.1 Data Mining Approaches

The recently developed depth sensors present some unique challenges. First, the depth maps are significantly noisy and the 3D positions of the tracked joints may be wrong if serious occlusions occur. This increases the intra-class variations in 3D actions. Second, depth sensors provide the opportunity to get a rich collection of features from multiple modalities. However, not all the features offer good discrimination among different actions. Also rich representations makes the computational requirements more expensive. It is expected that a more robust, effective and discriminative feature representation can be obtained using data mining approaches over rich collections.

For the action of a complex articulated structure, the motion of the individual parts are correlated. Wang et al. [20] propose a data mining solution to discover the discriminative conjunction rules. The idea is inspired by the successful applications of AND/OR graph learning in [21, 22]. An *actionlet* is defined as an AND structure within the base representation. A mining algorithm is developed in [20] to mine a set of discriminative actionlets. Chen et al. [23, 24] develop a mining approach to discover the key joints for 3D actions, which aims to capture the intra-class variance. They employ a multiple kernel learning algorithm [25] to learn the combinations of discriminative joints for each action class.

The above methods mine the discriminative information from high-level representations (e.g., joints, body parts), which does not guarantee good discrimination for the base representations (e.g., histogram-based descriptors). Not all prototypes in a histogram give the same level of discrimination among different action. Xia and Aggarwal [26] address this issue and mine discriminative feature sets from the feature pool based on F-score. They rank the prototypes by their F-scores and select features with high F-score. Oreifej and Liu [27] pay attention to the quantization methods to build histogram-based descriptors. The bins of the histogram are voted by using only the videos corresponding to the weighted set of support vectors from Support Vector Machines (SVM) classifiers.

2.2.2 Direct Modeling Approaches

Approaches that we discuss in this section do not take special attention of the temporal domain. They describe all frames of an observed sequence into a single representation or perform action recognition for each frame individually. In this case, the temporal variations of the actions are ignored. In Section 2.2.2.1, we discuss the discriminative classifiers which learn a function that discriminates between two or more classes by directly operating on the video representation. In Section 2.2.2.2, we discuss random forest (RF) method. RF is also considered as a combined method for fusion of features and action classification together. In Section 2.2.2.3, we discuss nearest neighbor classification where an observed sequence is compared to labeled sequences or action class prototypes.

2.2.2.1 Discriminative Classifiers

Discriminative classifiers, widely used for 2D action recognition, have also been employed to 3D data. Discriminative classification methods focus on separating two or more classes. One of the main methods of classification is SVM. SVM learn a hyperplane in feature space that is

described by a weighted combination of support vectors. SVM have been used in combination with fixed-length representations, such as bag of words features.

Recent discriminative classifiers for 3D action recognition have seen a growth trend in using Mutiple Kernel Learning SVM (MklSVM). It is expected that representations extracted from one modality complement the drawbacks of the representations extracted from other modalities, and combining several modalities increases the classification performance. MklSVM is considered as an ensemble method to compute the optimal linear combination of multi-modality/multi-type representations for the task of 3D action recognition. Wang et al. [20] employ MklSVM to learn a representation ensemble structure that combines the discriminative representations in the data mining step, where each kernel corresponds to a discriminative representation. Ofli et al. [28] learn a classifier by combining various modalities (e.g., RGB video, depth video, audio, accelerometer data) through MklSVMs, where each kernel corresponds to a representation extracted from one modality. Chen et al. [24] learn a linear combination of joint-based representations using Mkl, where each kernel corresponds to a joint-based representation.

2.2.2.2 Random Forest

Random forest (RF) is considered as a discriminative classifier using tree predictors in which each tree splits the data depending on the randomly selected features. RF chooses the most popular class label which obtains the most votes over all trees. They are widely used as multi-class classifiers [29, 5], especially in 2D action recognition due to several nice properties of RF (e.g., robustness to noise and efficiency for classification). These properties are also admired by the 3D action recognition task.

As a RF [30, 31] is an ensemble of weak learners which are decision trees, several authors [32, 33, 34] have used RF as classifiers in combination with feature selections and feature fusion. Rehmani et al. [33] employ RF for feature selection and 3D action classification together. RF is first trained using the set of all proposed features in [33] and then all the features whose scores are below a specified threshold are deleted. The new compact feature vectors are then selected to train a new RF to be used for classifications. As discussed earlier, Zhu et al. [34] use RF for fusion of distinct features and action classification together.

2.2.2.3 Nearest Neighbor Classification

Nearest neighbor classifiers (NN) compute the distance between the video representation of the observed sequence and those in the training set. The effectiveness of NN for 3D action classification largely depends on the computation method of the distance. One of the widely used methods is k-Nearest Neighbor classifiers (kNN). kNN compute the distance between the observed sequence and each video in the training set [35]. The most common label among k closet training videos is chosen as the classification result. Ballin et al. [36] use the 1-Nearest neighbor classifiers for 3D action recognition. The label related to the 3D video representation that has the shortest distance to the test video is chosen as predicted class. For a large training set, such comparisons are computationally expensive. Recent work by Boiman et al. [37] observe that computing image-to-class distance, which depends on the distribution of the representation over the entire class, provides a better generalization capability than image-to-image distance. They develop the Naive-Bayes-Nearest-Neighbor (NBNN) as classifiers for image classification. Inspired by [37], Yang and Tian [38] extend these concepts of NBNN-based image classification to NBNN-based 3D action recognition. They directly use single frame representations without quantization, and compute video-to-class distance rather than video-to-video distance. Seidenari et al. [39] follow this method, and use NBNN classifier to score the similarity of a query 3D action video and predefined action class.

2.2.3 Temporal State-based Approaches

So far, the majority of works in 3D action recognition have not considered temporal modeling as a final stage in the classification process. This is due to the fact that existing approaches mainly employ static methods, or encode the temporal dimension into the video representation. However, several temporal models are used in 2D action recognition in order to model the dynamics as part of the classification process. This has been followed in a number of recent works in 3D action recognition.

Temporal approaches classify a human action by a temporal model composed of a set of states. These states are connected by edges, which model probabilities between states, and between states and observations. Each state describes the action performance at a certain moment in time. An observation corresponds to the representations at a given time. The temporal model is learned so that it corresponds to the 3D video representations belonging to its action class. As temporal approaches used by recent methods for 3D action recognition are

generally similar to those used for the 2D case, we briefly present these methods. In Section 2.2.3.1, we discuss Dynamic time warping (DTW), which is a distance measure between two 3D videos with different lengths. In Section 2.2.3.2, we discuss Hidden Markov Models (HMM) for temporal modeling, where learn a joint distribution over both, observations and action labels. In Section 2.2.3.3, we discuss the action graph classifier, which is regarded as a generative model and is able to perform on-line 3D action recognition.

2.2.3.1 Temporal Warping Approaches

Temporal warping approaches learn to align two action sequences and measure their matching scores. They simultaneously take into account a pair-wise distance between corresponding frames and the sequence alignment cost [40, 41]. The key points of temporal warping approaches are the accuracy and robustness of action recognition under temporal misalignment and duration variation in video actions. Dynamic time warping is a distance measure between two sequences with different lengths. It is widely used in 2D action recognition, and dynamic programming is often used to calculate the optimal alignment [40, 41]. In [42, 43], they propose a 3D action recognition system based on DTW and bag of key poses. An equivalent sequence of key poses is obtained for the test video. DTW is used to find the most similar training sequence considering a temporal alignment of the involved key poses. Reyes et al. [44] present a 3D gesture recognition approach based on DTW framework. Depth features from human joints are compared through video sequences using DTW, and weights are assigned to features based on inter-intra class gesture variability. Similarly, [45, 46] use DTW to tackle the temporal length variability of 3D gesture. Bautista et al. [46] develop a probability-based DTW for 3D gesture recognition. Different samples of the same gesture pattern are used to build a Gaussian-based probabilistic model. The cost of DTW is adapted to the model. Wang and Wu [47] develop a discriminative learning-based temporal alignment method, maximum margin temporal warping (MMTW), to align two action videos and measure their matching scores.

2.2.3.2 Hidden Markov Models

Hidden markov models (HMM) are regarded as a generative approach which learn a joint distribution over both observations and action labels. HMM use hidden states that correspond to different phases in the representation of an action. They model state transition probabilities and observation probabilities. HMM have been used in 2D recognition tasks in a large number of works [48, 49, 50], due to their suitability for modeling pattern recognition problems that

exhibit an inherent temporality [51]. Gaschler et al. [52, 53] train HMM model using human body posture and head pose estimation from depth cameras to recognize human social behaviors. Xia et al. [54] recognize 3D human actions by the discrete HMM [51]. They encode each action sequence as a vector of posture vocabularies, and learn the HMM model to predict the class label of the test video. Dubois and Charpillet [55] create HMM with eight states corresponding to eight action classes. The same approach has also been taken by [56, 57].

2.2.3.3 Action Graphs

Instead of performing off-line recognition, the action graph classifier which is proposed by Li et al. [58] has the advantage that it can perform classification without having to wait until an action is finished. An action graph is a system composed of a set of trained action classes, a set of key poses, an observation model, and a set of matrices to model the transition probability between key poses. Given a test 3D video represented by a sequence of short-temporal features, the probability of occurrence of the sequence with respect to each trained action class is computed. Similar to HMM, the action graph model is flexible in handling performance speed variations, and takes into account the temporal dependency. Compared to HMM, action graph models have the advantage that they require less training data and allow different actions to share states [59]. Vieira et al. [59] and Kurakin et al. [60] extend the action graph classifier from 2D video recognition to 3D cases. They employ the action graph to 3D action recognition and 3D dynamic hand gesture recognition, respectively.

2.2.4 3D Action Detection and Anticipation

Action detection and anticipation techniques are aiming at detecting the past and anticipating future actions, respectively. Approaches do not explicitly model the video representation of the actions, nor do they model the action dynamics. They correlate an observed sequence to labeled video sequences. These works have several dissimilarities to those previously discussed (e.g., 3D action classification in Section 2.2.2 and Section 2.2.3). The methods used for 3D action classification only address recognition over small periods of time, where temporal segmentation is not a big problem (e.g., temporal segmentation has been done apriori). While action detection and anticipation consider joint segmentation and classification by defining dynamical models [61, 62, 63]. We discuss 3D action detection and anticipation in Section 2.2.4.1 and Section 2.2.4.2, respectively.

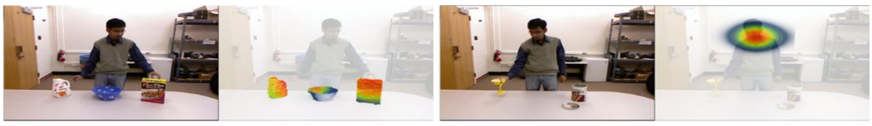
2.2.4.1 Action Detection

Action detection requires the system to not only recognize but also localize certain actions of interest in a video sequence spatially and temporally. Action detection is a difficult problem due to the individual variations of people in posture, changes in view angle, and the complex and cluttered background. As the depth information provided by 3D video induces three dimensional contextual information for modeling interactions, it promotes recent methods to tackle this problem.

Vieira et al. [59] develop an on-line 3D action recognition system based on action graph [58] and an SVM pose classifier. The SVM pose classifier is trained using the short-temporal features in oder to address temporal segmentation. The action graph performs the 3D action recognition. In the work of [64, 65, 66] present a privacy preserving automatic fall detection method to facilitate the independence of older adults. More recently, in [67, 68], they performed detection and recognition of unstructured human action in unstructured environments. The method is based on a hierarchical maximum entropy Markov model [69]. Ni et al. [70, 71] propose a complex action recognition and localization approach. A latent structural support vector machine model [72, 73] is developed to fuse the information from multiple levels of 3D video processing and form a discriminative 3D action detection framework.

2.2.4.2 Action Anticipation

Action anticipation considers the problem of anticipating which actions will happen in the future and how. Action anticipation is interrelated with action detection, and can improve the performance of the detection accuracy of past actions [74]. It has gained attention only recently [75, 76, 74, 77] (see Fig 2.1). Koppula et al. [75] provide a method to learn human actions by modeling the sub-actions and affordances of the objects. They define a Markov random fild (MRF) which is built with each spatio-temporal segment being a node. The model is learnt using a latent structural support vector machine [72, 73] approach. Koppula and Saxena [74] address the problem of anticipating human actions at a fine-grained level of how humans interact with objects in more complex actions. They represent the distribution of the possible futures with a set of particles that are obtained by augmenting the MRF structure of [75] with sampled future nodes. The above methods focus on performing learning and inference given the spatio-temporal structure of the model (e.g., a given conditional random field structure in [75]), which is quite challenging to estimate. Because of the ambiguity in the temporal segmentation, a single graph structure is not enough to explain the action well. This issue is addressed by [76],



(a) Affordance Heatmaps

(b) Anticipated Trajectory Heatmaps

(c) Robot Anticipatory Response

Figure 2.1: Action anticipation: (a) The first two images show the reachability affordance heatmaps and the last two show the drinkability affordance heatmap, (b) The heatmap of anticipated trajectories for the action and how the trajectories evolve with time, (c) Robot anticipatory response for refilling water task [75](best viewed in color).

who propose a method to obtain potential graph structures that are close to the ground-truth ones.

Chapter 3

Unsupervised Feature Learning Framework with RGBD Data

In this chapter, we will present our unsupervised feature learning framework with RGBD video and skeleton data. We will firstly discuss the state-of-art techniques employed for 3D video representations from RGBD sensors. Ideal representations should generalize over small variations in background, viewpoint and action execution. At the same time, the representation should be sufficiently rich for 3D human motion perception tasks.

3D video representations in human motion perception can be roughly separated into three categories: extended 2D-based representation, depth-based and representation, and skeleton-based representation. These classes are introduced to address the typical characteristics for the corresponding modalities (e.g depth video data, skeleton data). In Section 3.1, we discuss extended 2D-based representations which extend the features based on the 2D video data (e.g., hand-designed features like STIP [78]) to the depth video data. In Section 3.2, we discuss depth-based representations which represent actions utilizing depth information directly. In Section 3.2.3, we discuss skeleton-based representations, where human actions are represented by skeleton informations. All these sections that describe the 3D video representation methods contain discussions about their respective advantages and disadvantages.

We will finally present our unsupervised feature representation framework with RGBD video and skeleton data in Section 3.3. Unsupervised learning of features captures discriminative and hierarchical data information by learning multiple layers of representations via artificial intelligent models. Nowadays, there is a growing interest in unsupervised feature learning methods such as Deep Belief Nets [79, 80, 81], Sparse Coding [82, 83], Convolutional Neural

Networks [84], Independent Component Analysis (ICA) and Independent Subspace Analysis (ISA) [85]. These biologically-inspired learning algorithms show promise in the domain of computer vision and autonomous driving, such as object recognition with RGB-D images [86], action recognition with RGB video data [84], speech recognition [87], and end to end learning of driving behavior in autonomous driving [88]. Although many of work exist for learning features from RGB image or RGB video data, none has yet been investigated for depth video and skeleton data. In this section, we provide an unsupervised feature learning framework with RGBD video and skeleton data inspired by [6, 85]. We describe our approach for learning 3D spatio-temporal feature from RGBD video data and 3.5D spatio-temporal feature from RGBD video and skeleton data in Section 3.3.2 and Section 3.3.3 respectively. As a general framework, it can be used by any discriminative and generative classifiers in human motion perception tasks.

3.1 Extended 2D-based Representations

Extended 2D-based representations are extracted from the 3D video data, using extensions of traditional 2D-based methods. These methods are widely used in action recognition from 2D videos and provide a compact representation of 2D video data by describing the appearance, geometry, and motion of local parts of the video data.

To exploit the additional information (e.g., structure information) in the depth data, a few extended algorithms based on the existing 2D-based methods [89, 90, 91, 92, 93] were recently proposed. The majority of these algorithms focus on local representation of 3D video data. Local representations describe the human actions as a collection of independent descriptors. The calculation of local representations proceeds in a bottom-up fashion: interest point extraction, feature description and final representations. We discuss the interest point detection and description in Section 3.1.1 and Section 3.1.2 respectively. A small number of works report the use of global representations for 3D action recognition. Global representations are obtained in a top-down fashion: background subtraction, subject localization, and image description. First, subject regions are obtained through background subtraction or tracking. Then, the subject regions are encoded as a whole, which result in the final representations. We discuss these briefly in Section 3.1.3.

3.1.1 Spatio-temporal Interest Point Detection

Spatio-temporal interest points are the locations in space and time where interesting motions occur in the video. It is assumed that these points are the most informative for the recognition of human actions.

The simplest way to calculate spatio-temporal interest points is by directly applying the spatio-temporal detectors (e.g., 3D Harris detector [89]) to the depth video data. Hernandez-Vela et al. [45] apply the Harris detector separately on the RGB and depth video data. One drawback of this method is that it treats the depth video as gray-scale video data without using the spatial information along the depth direction. These issues are addressed by Oreifej and Liu [27], who conducted an experiment applying local interest point detectors [89] on the MSRDaily3D dataset [20]. It is not superising that the detected interest points are fired on locations irrelevant to the action of interest. To address this, several approaches are proposed. The most straightforward way is to utilize the depth information to perform the feature pooling by dividing the entire scene into different depth layers. Ni et al. [94] use the depth value of each detected interest point to divide the entire depth range into a set of depth layers.

In contrast to the above approach, where depth information is ignored during interest point extraction, recent work by Hadfield and Bowden [95] incorporate the depth information while detecting the spatio-temporal interest points. They extend the 3D Harris detectors [89] and the 3D Hessian detectors [90] to 4D cases by exploiting the relationship between the spatio-temporal gradients of the depth stream and those of the appearance stream. In part, the 3D Harris detectors are motivated by the idea that object boundary points are highly salient. As depth data directly provide boundary information, rendering the estimation of the intensity gradient along depth direction is somewhat redundant. Hadfield and Bowden [95] extend the Separable Filters [91], Harris and Hessian detectors to 3.5D by exploiting complimentary information between the appearance and depth streams.

3.1.2 Feature Descriptors

Feature descriptors summarize an image or video patch in a representation that is ideally invariant to background clutter, appearance and occlusions, and possibly to rotation and scale. The spatial and temporal size of a patch is usually determined by the scale of the spatio-temporal interest point.

Similar to spatio-temporal interest point detection, one straightforward way is applying the common feature descriptors (e.g., HOG [92]) on the depth video data. Zhao et al.

[96] and Ni et al. [94] use the HOG [92] and HOF [93] features to describe the interest points on the depth video data. Instead of appearance, motion and saliency be used in above descriptors, depth information can be utilized. Hadfield and Bowden [95] extend the HOG and HOF to 4D descriptors (HODG), which encapsulate local structural information, in addition to local appearance and motion. Hadfield and Bowden [95] extend the RMD (Relative Motion Descriptor) [97] to 4D (RMD-4D). RMD-4D make use of saliency information within a 4D integral hyper-volume during interest point detection.

3.1.3 Global Feature Representations

Global feature representations encode the region of interest (ROI) of a subject as a whole. The ROI is usually obtained through background subtraction or subject tracking. Common global representations are derived from silhouettes, edges or optical flow.

One widely used global feature representation method for 2D action recognition is motion history images (MHI) developed by Bobick and Davis [98]. MHI are capable of encoding the dynamics of sequences of moving human silhouettes. A MHI is constructed where each pixel intensity is a function of the motion recency at that location. However, due to using 2D cameras, MHI can only encode the history of motion induced by the lateral component of the scene motion parallel to the image plane. With the depth information, Ni et al. [94] develop 3D-MHI which encode the motion history along the depth changing directions. 3D-MHI consist of three images: fDMHI, bDMHI, MHI. fDMHI encode the forward motion history of moving human silhouettes while bDMHI encode the backward motion history of moving human silhouettes. Yang et al. [99] encode the motion history from front, side and top views to make use of the additional body shape and motion information from depth maps.

3.2 Depth-based and Skeleton-based Representations

Depth-based representations describe the observation using features extracted from depth video data. Although depth data have existed for several decades, depth-based representations have stayed at the level of describing static scenes or objects for a long time [100, 101, 102, 103]. Existing features aim at object recognition, place recognition or 3D registration. Compared to 2D video data, depth video data provide geometry and shape cues. These advantages as well as the recent advent of low-cost depth sensors motivate researchers to seek new representations based on depth video data.

The fundamental difference between extended 2D-based representations and depth-based representation is that the latter designs features to characterize the unique properties of the depth video data directly, instead of extending existing algorithms based on 2D video data. Similar to extended 2D-based representations in Section 3.1, we divide depth-based representations into two categories: local depth-based representation and global depth-based representation, which are discussed in Section 3.2.1 and Section 3.2.2, respectively.

3.2.1 Local Depth-based Representations

Local depth-based representations describe human actions as a collection of local descriptors extracted from depth video data. As depth video data capture the 3D spatial structure of subjects, the local depth-based representation is designed to explore the embedded structural relations for action analysis.

Hernandez-Vela et al. [45] extend the VFH feature [102] from pose estimation domain to 3D human action recognition domain. As the VFH feature is invariant to the rotation about the roll axis of the camera, [45] encode the roll axis information by creating a camera roll histogram (CRH). The final feature is the concatenation of VFH histogram and CRH histogram. Cheng et al. [104] design the comparative coding descriptor (CCD) to capture the spatial geometrical relations and related variations over time. The CCD feature uses the comparative description idea in Local Binary Patterns [105]. Inspired by the CCD and LBP features, Zhao et al. [96] develop the local depth pattern feature (LDP). A local region is partitioned into spatial cells. The average depth value is computed for each cell. The difference of average depth values between every cell pair form the LDP feature.

Wang et al. [106] treat depth videos as a 4D volume. They employ four dimensional random occupancy patterns to construct the features (ROP). The ROP features capture the occupancy pattern of a 4D subvolume. The self-similarity based feature has been proven to work well at object detection and action detection on RGB video data [107]. Xia and Aggarwal [26] adopt the idea of [107] and build a depth cuboid similarity feature (DCSF) to describe the local 3D depth cuboid around the interest points with an adaptable supporting size. This feature captures characteristic shapes and motion.

3.2.2 Global Depth-based Representations

Instead of using local features to represent the depth video data, global depth-based representations are obtained from the entire depth video data. They are robust against noise, occlusions

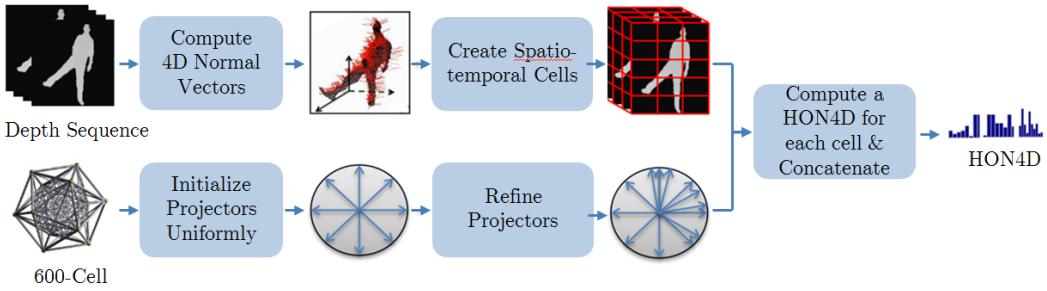


Figure 3.1: The process of computing the histogram of oriented 4D surface normals (HON4D) [27].

and variations from which local methods suffer.

By partitioning the ROI into a fixed dimensional grid, small variations due to noise, partial occlusions and changes in viewpoint are partially overcome. These grid-based representations are employed by RGB action recognition [108, 109, 110, 111, 112]. Several global depth-based representations proceed along this direction. Vieira et al. [59] develop the space-time occupancy patterns (STOP) as a global depth-based representation from depth video data. The space and time axes are divided into multiple segments to define a 4D grid for each depth video. This preserves spatial and temporal contextual information between space-time cells while being flexible enough to accommodate intra-action variations.

As discussed in Section 3.1.3, shape, edges, and motion information are commonly used in global 2D-based representations. Because the depth maps provide natural surfaces to capture the geometrical features of the observed scene, Oreifej and Liu [27] use the normal vectors in global depth-based representations instead of the gradients in global 2D-based representations. They describe the depth video data using a histogram of oriented 4D surface normals (HON4D) (see Fig. 3.1). HON4D is analogous to the histograms of gradients in RGB sequences [92, 113], and extends the histogram of normals in static depth images [114].

3.2.3 Skeleton-based Representations

Skeleton-based representations describe actions in depth video data by modeling the spatio-temporal structures of the human skeleton. It is first introduced by Johansson [115, 116]. This paradigm has been further studied in [117, 118, 119, 120, 121]. Compared to extended 2D-based representation and depth-based representation, skeleton-based representations model temporal dynamics and spatial structures explicitly. The skeleton has a natural correspondence

across time. It is non-trivial to quickly and reliably extract and track accurate skeletons from 2D videos. Motion capture systems provide clean skeleton data but they are rarely used due to the high cost of the systems [122, 123, 124]. Multi-view RGB image capture systems produce stable skeleton estimations but it is limited to only laboratory environments due to its complex settings [125, 126, 127]. A recently developed human skeleton capturing system by Shotton et al. [5] is able to recover 3D position of skeleton joints in real time and with reasonable accuracy [128, 129, 130]. It has motivated recent research work to investigate 3D action recognition using the skeleton information [38, 131].

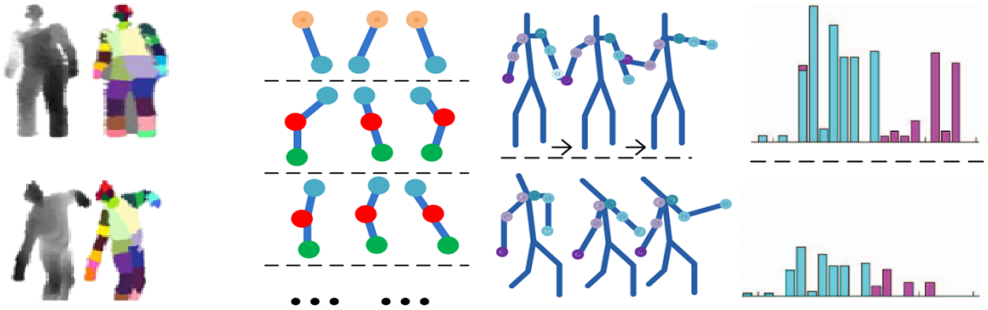
In this section, we focus on skeleton-based representations using skeleton joints extracted from depth video data. We divide skeleton-based representations into two categories: short-term posture-based representations and long-term trajectory-based representations, and discuss these two in Section 3.2.3.1 and Section 3.2.3.2, respectively.

3.2.3.1 Short-term Posture-based Representations

Short-term posture-based representations represent the posture and motion features using only skeleton joints which encode the spatial and temporal structures in single frame or consecutive frames.

Yang and Tian [38, 132] develop the EigenJoints features based on differences of skeleton joints. EigenJoint is able to characterize action information including static posture features, consecutive motion features and offset features in each frame. Bloom et al. [133] use pose-based features such as position difference, position velocity, position velocity magnitude, angle velocity and joint angles. Yun et al. [134] introduce six types of relational body-pose features (the joint distance feature, the joint motion feature, the plane feature, the normal plane feature, the velocity feature, and the normal velocity feature) which are also used in [135, 136]. The relational body-pose features describe geometric relations between specific joints in a single pose or a short sequence of poses (e.g., 2-3 frames).

Xia et al. [54] develop the histogram of 3D joints (HOJ3D), a viewpoint invariant representation of postures based on 3D skeleton joint locations. They compute HOJ3D from 12 of 20 joints to the exclusion of redundant informations. Inspired by recent findings from neuroscience in the work of [137, 138], Chaudhry et al. [139] develop the dynamic medial-axis features. They use a spatio-temporal hierarchy of skeleton configurations instead of modeling the entire human skeleton using a single feature representation.



(a) Skeleton **(b)** Dictionary **(c)** Part Sets **(d)** Action

Figure 3.2: The framework for action representations: (a) Skeleton data, (b) Dictionaries of part poses, (c) Temporal-part-sets and spatial-part-sets, (d) Action response in [141](best viewed in color).

Several works have explored the feature learning strategy of combining frequently co-occurring primitive features into larger compound features (e.g., [140, 20]). The *low-level* features such as optical flows are commonly used instead of *high-level* features [140]. Wang et al. [141] use the pose-based features for action recognition. They apply data mining techniques [142] to obtain sets of distinctive co-occurring spatial and temporal configurations of body parts (see Fig. 3.2).

3.2.3.2 Long-term Trajectory-based Representations

Long-term trajectory-based representations interpret an action as a set of space-time trajectories. In long-term trajectory-based representations, a person is generally represented as a set of local features (e.g., joint angles) corresponding to the time-series 3D joint positions. The early work done by Johansson [115] suggested that the tracking of joint positions is itself sufficient for humans to distinguish actions [7]. The skeleton tracker [5] allows people to know exactly the correspondences between the joints on any two skeletons. This natural characteristic stimulates the research in the domain of trajectory-based approaches.

Raptis et al. [143] design an angular representation of the skeleton. They fit the full torso with a single frame of reference, and use this frame to parameterize the orientation estimates of both the first and second degree limb joints. Seidenari et al. [39] propose a skeletal based representation based on the work of [143]. Differently, Seidenari et al. [39] represent the coordinates of the first degree joints in Cartesian coordinates while Raptis et al. [143] represent them in polar coordinates.

Gowayyed et al. [144] propose a 2D trajectory descriptor, histogram of oriented displacement (HOD). Given a sequence of joints locations, they describe the 3D trajectory of each individual joint. Unlike the common representations representing the evolution of the time series of all joints, Ofli et al. [145] consider sequences of the most informative joints as a new feature representation. The final representations represent a sequence by encoding set of the most informative joints at a specific time instant as well as the temporal evolution of the set of the most informative joints through out the trajectory.

The covariance descriptor was introduced by Tuzel et al. [146], and was applied successfully on object detection and texture classification [146, 147]. Sanin et al. [148] first apply this idea to 2D action recognition by considering the feature of pixels in a spatio-temporal patch. Inspired by the finding of [148, 136], Hussein et al. [149] use body joint locations, sampled over time, as the variables on which covariance descriptors are computed.

3.3 Efficiency with Unsupervised Feature Learning Framework

3.3.1 Unsupervised Feature Learning vs. Supervised Feature Crafting

RGBD cameras provide rich modalities, such as skeleton joint data, depth data, gray-scale data, surface normal data. A key question arises when using RGBD camera: how can we represent rich data in an efficient way? State-of-the-art techniques represent video data by extracting hand crafted features, which we call supervised feature crafting. Hand-crafted features like STIP [89], or use the local features like HOD [144] or HOG [92] are designed by professional experts to represent the interesting spatio-temporal patterns of the video data. Despite their good performance for 3D human motion recognition, these methods suffer from several problems. Firstly, the heavy workload of designing features by hand is inefficient. People need to have enough professional knowledge about the internal properties of the modality to design new features. The only way to design a good feature is by trial and error manually. Another drawback of state-of-the-art hand-designed features is that they are highly problem-dependent. This issue is addressed by Wang et al. [6]. They found that there is no universally best hand-engineered feature for different datasets. Also, data dependent is another issue: the depth modality provides useful extra information to the complex problem of activity recognition since depth information is invariant to lighting and color variations. However, because there is no texture in the depth data,the extending hand-designed features from RGB data to depth

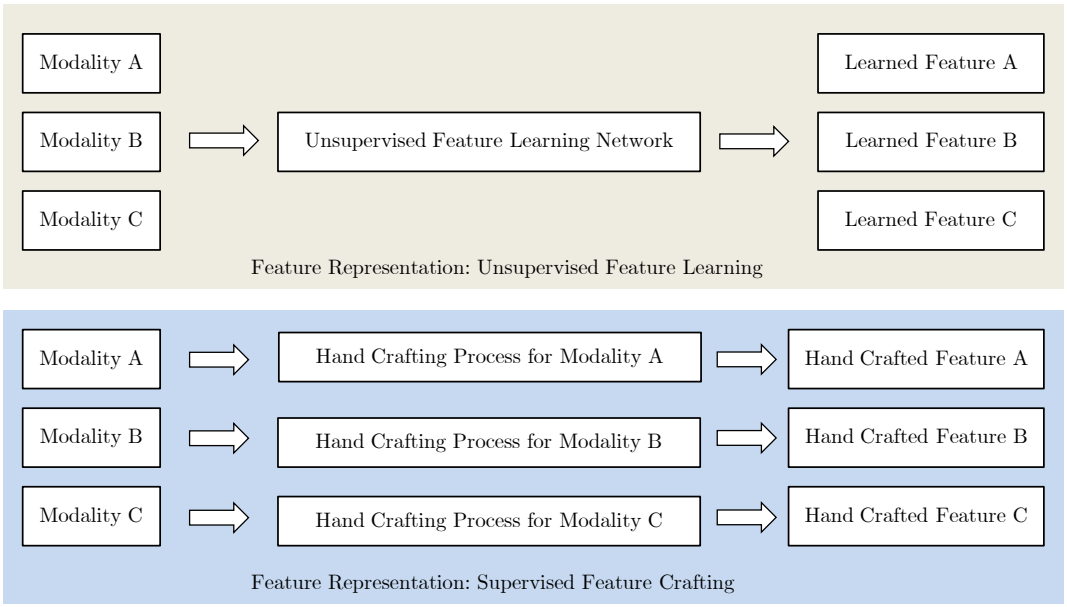


Figure 3.3: Unsupervised feature learning vs. supervised feature crafting.

modality are not discriminative enough for classifications. In addition, the depth video data is full of noises. There are large shadows or holes in the depth data. Lastly, it's the new modality issue: skeleton joint data is a new type data provided by RGBD cameras. Different with the traditional dense video data by capturing sequence of 2D frames with RGB or depth channels, sparse skeleton data have natural correspondences over time, which model temporal dynamics and spatial structures explicitly. Spatial structure is important to many computer vision tasks, e.g, spatial pyramid match for image recognition. Together with other data modalities (e.g. depth video data), skeleton data may improve the performance on 3D action recognition. All these issues make the RGBD feature development of the state-of-the-art techniques inefficient. Actually, these issues are also very common in other computer vision tasks. A well-know technique in image and 2D video recognition to tackle these issues is called unsupervised feature learning. The major advantage of unsupervised learning models is that it leverage the plethora of unlabeled data and adapt easily to new modalities. Fig. 3.3 is a comparison of unsupervised feature learning and supervised feature crafting. Unsupervised learning models [150, 85] are a class of machines that automatically build high-level representations of unlabeled input from low-level ones (e.g., pixels) without pre-processing steps. To understand the contribution of unsupervised learning approaches, Le et al. [85] follow experimental protocols of [6] by using

their standard processing pipeline and only replacing the first stage of feature extraction with deep learning method. Le et al. [85] use an extension of the independent subspace analysis algorithm to learn invariant spatio-temporal features from unlabeled video data. Their method performed consistently better than a wide variety of combinations of methods. Ji et al. [84] extend the convolutional neural network to the 2D video data and achieve good performance on human action recognition task. Based on the convolutional Restricted Boltzmann machine (CRBM) [79, 80, 151], Chen et al. [81] build a hierarchical, distributed model for unsupervised learning of invariant spatio-temporal features from video. Their model, called the Space-Time Deep Belief Network (ST-DBN), alternates the aggregation of spatial and temporal information so that higher layers capture longer range statistical dependencies in both space and time. The ST-DBN has superior feature invariance properties and is able to integrate information from both space and time to fill in missing data in video.

3.3.2 3D Spatio-temporal Feature Learning from RGBD Video Data

We firstly provide an unsupervised learning method to learn 3D representations from RGBD video. At the heart of our method is the application of Independent Subspace Analysis (ISA). The ISA algorithm is a well-known algorithm in the field of natural image statics [152]. Experimental studies have shown that this algorithm can learn very discriminative features from static images or color-based sequences. A main advantage of ISA is that it learns features that are robust to local translation, while being selective to rotation and velocity. We extend the original ISA algorithm for the use of four different channels of video data independently, namely grayscale modality data, gradient modality data, depth modality data and surface normal modality data. Fig. 3.4 outlines our approach.

ISA is an unsupervised learning algorithm that learns features from unlabeled subvolumes. First, we extract random subvolumes from different modality types of video data. We then normalize and whiten the set of subvolumes. We feed the pre-processed subvolumes to ISA networks as input units. An ISA network [152] is described as a two-layer neural network, with square and square-root nonlinearities in the first and second layers respectively.

We start with any input unit $x^t \in \mathbb{R}^n$ for each random sampled subvolume. We split each subvolume into a sequence of image patches and flatten them into a vector x^t with the dimension n. The activation of each second layer unit is

$$p_i(x^t; W, V) = \sqrt{\sum_{k=1}^{m} V_{ik} \left(\sum_{j=1}^{n} W_{kj} x_j^t\right)^2} \tag{3.1}$$

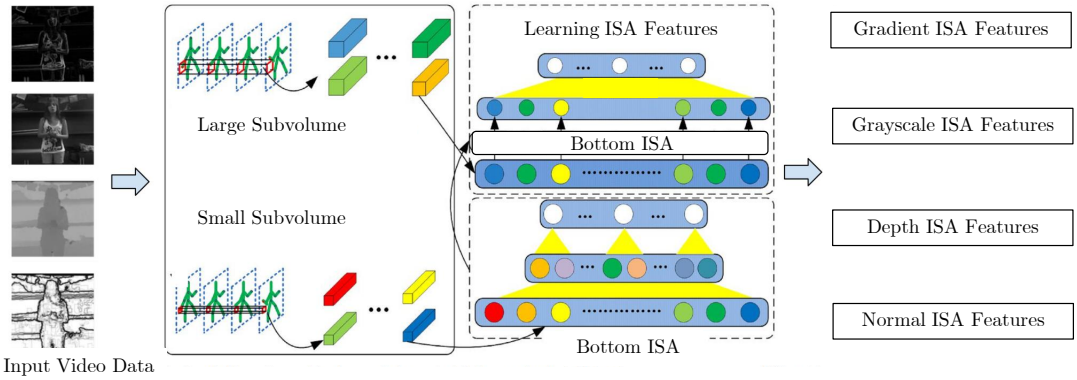


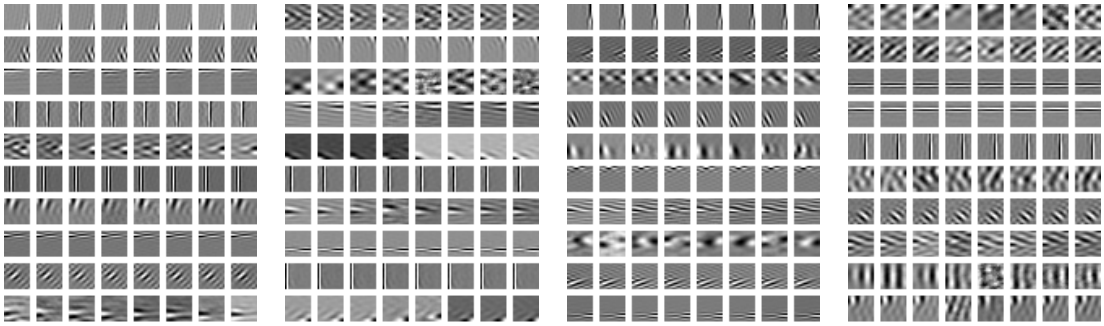
Figure 3.4: An overview of our extended ISA model: we randomly sample subvolumes from different modality types of video. These subvolumes are given as input of the two-layers ISA network. Our ISA model learns four types of features: Gradient ISA feature, Grayscale ISA feature, Depth ISA feature, and Surface Normal ISA feature (best viewed in color).

ISA learns parameters W through finding sparse feature representations in the second layer by solving

$$\begin{aligned} &\min_{W} \sum_{t=1}^{T} \sum_{i=1}^{m} p_i(x^t; W, V) \\ &s.t. WW^T = \mathbf{I} \end{aligned} \tag{3.2}$$

Here, $W \in \mathbb{R}^{k \times n}$ is the weight connecting the input units to the first layer units. $V \in \mathbb{R}^{m \times k}$ is the weight connecting the first layer units to the second layer units; n, k, m are the input dimension number of the first layer units and second layer units respectively. The orthonormal constraint ensures feature diversity.

As is common in neural networks, we stack another ISA layer with PCA on top of the bottom ISA. We use PCA to whiten the data and reduce the dimensions of the input unit. The model is trained greedily layerwise in the same manner as other algorithms described in [150, 85]. The model so far has been unsupervised. The bottom ISA model learns spatio-temporal features that detect a moving edge in time as shown in Fig. 3.5. We learn spatio-temporal features up to four channels of RGB-D video data: grayscale, gradient, depth, and surface normal (z-axis). These features are interesting to look at and share some similarities. For example, the learned feature (each row in Fig. 3.5) is able to assign similar features in a group thereby achieving spatial invariance. The features have sharper edges like Gabor filters.

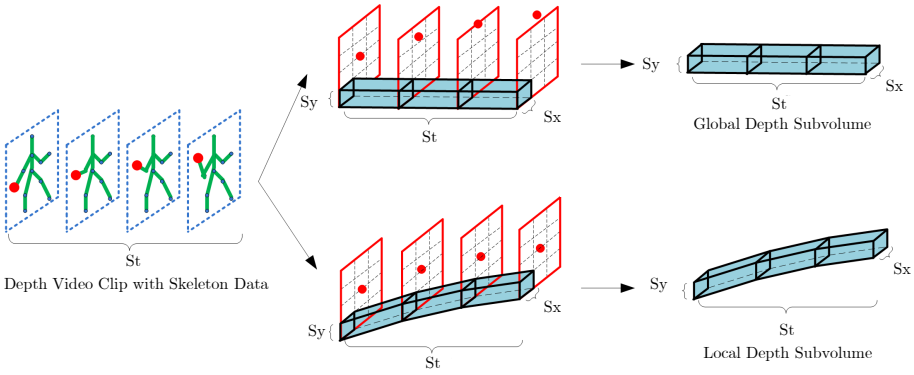


(a) Grayscale **(b)** Depth **(c)** Grayscale **(d)** Surface Normal

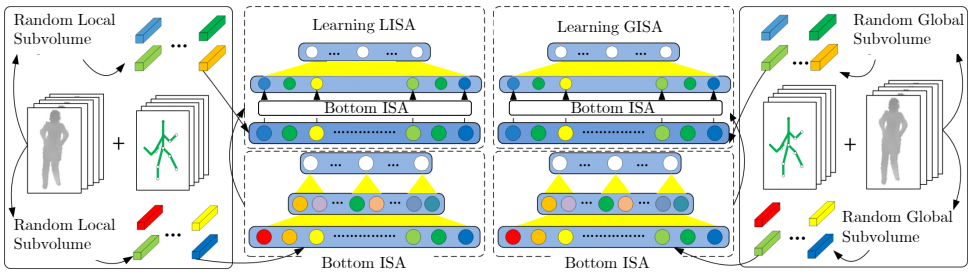
Figure 3.5: Visualization of randomly selected spatio-temporal features learned from four channels of the RGB-D video data - from left to right, grayscale, depth, gradient magnitude, Z surface normal component.Each row of the figure indicates a spatio-temporal feature

3.3.3 3.5D Spatio-temporal Feature Learning from RGBD Video and Skeleton Data

A disadvantage of ISA is that it can be rather inefficient to train with high dimensionality data. Here we extend our work of Section 3.3.2 by integrating the skeleton joint date into the feature learning process. Instead of training the model with the entire video, we apply the ISA algorithm to local regions of joints to improve the training efficiency. We name it Joint-ISA model (See Fig. 3.6). Based on the video data and the estimated 3D joint positions, we develop two types of spatio-temporal features: Global ISA features (GISA) and Local ISA features (LISA). We define GISA features as the global spatio-temporal features learned by the stacked Joint-ISA model. Given a global subvolume (e.g., a sequence of depth patches), the depth patches of the subvolume have the same spatial positions under the image coordinate of the depth video across time. Similarly, LISA features are learned by the stacked ISA model with input: the local depth subvolumes. The different depth patches of the local subvolumees have the natural correspondences. It is able to track the depth patch one by one across time. These spatio-temporal features can be treated as the resulting descriptors of spatio-temporal interest points. Intereset points are densely sampled from the surrounding regions of the joints. We call GISA and LISA 3.5D spatio-temporal features referring to combined information of spatio-temporal features and the skeleton data (3D joint positions).



(a) Generate global and local subvolumes



(b) Learn two types of ISA features

Figure 3.6: Overview of our Joint-ISA model: (a) We randomly sample global depth subvolumes and local depth subvolumes from the surrounding regions of each joint. (b) these two types of subvolumes are given as inputs to the two-layer ISA network. Our ISA model learns two types of features: Global ISA feature (GISA) and Local ISA feature (LISA) (best viewed in color).

3.4 Discussion

RGBD cameras are very different from traditional visible light cameras. This has in turn led to the requirement to revisit problems such as object detection and action recognition using RGB-D video data, especially depth video data and skeleton data. RGBD sensors have several advantages. For example, depth images provide 3D structural information of a scene, which can often be more discriminative than color and texture in many applications including detection, segmentation and action recognition. Moreover, the depth information is invariant to lighting and color variations. These advantages have facilitated a rather powerful human motion capturing technique [5] that generates 3D joint positions of the human skeleton. All these new modality data require novel video representations to be developed in order to fully exploit the useful and powerful information. Traditional ways of creating video representation have

problem when they are directly applied to new modality data. People need expert knowledge of the new modality data. The way of creating a new type of video representation is also time-consuming and expensive. In contract, our method interpret the raw data (e.g., the pixels of the video data) automatically and directly without human intervention. This indeed avoids the laborious process of hand-crafting features. Besides, our method shares the same learning architecture with different input modalities, which makes the adaptation to new sensor data much easier. The major advantage of unsupervised learning framework is its suitability for big data, especially when the labels of the dataset are not available or annotating the huge data is not practical. For example, there is a huge dataset which only provides a small part of ground truth information due to tediously manual annotation. In this case, our method is perfect as the feature learning does not require any label of the training data. However, when the dataset is really huge, the learned filters need to be complete in order to capture enough information and achieve satisfied performance. This will make the feature set bigger, training time longer and the computation more expensive. We tackled this problem by integrating skeleton joint data into the feature learning process. In human motions, skeleton joint data can be treated as useful and discriminative information comparing to the entire video data. This makes the feature training efficient without loss of discrimination.

Apart from the advantages of unsupervised learning, there are disadvantages. Unsupervised learning cannot make use of additional data in an efficient way which makes the training tedious whenever there are new data coming. Because unsupervised learning are unguided, the interesting patterns discovered by itself may be completely uninteresting. This is also part of the reason that high-level recognition tasks often deploy supervised learning approach.

Chapter 4

3D Human Action Recognition With Un-supervision and Ensemble Learning

In 3D human motion perception, two significant questions arise when using depth cameras. First, how can we represent depth video data efficiently? State-of-the-art techniques represent depth video data by extracting manually designed features, either directly from depth video data, or extending hand-designed features from color-based video data [89, 92]. We addressed this problem with the help of our unsupervised learning framework in Chapter 3. Second, how can we deal with noisy human skeleton data and improve the robustness of 3D action recognition systems? Skeleton data have natural correspondences over time, which model temporal dynamics and spatial structures explicitly. Together with other modalities (e.g. depth video data), skeleton data may improve the performance of 3D action recognition. However, when skeleton data contain irrelevant or redundant information, performance may be adversely affected. In order to tackle the problem of tracking errors in skeleton data and to handle intra-class variations more robustly, we propose *an ensemble learning approach with discriminative multi-kernel learning* (EnMkl) for 3D action recognition. The general framework of our proposed approach is show in Fig. 4.1. In our implementation, we formulate the 3D action recognition task with depth video as a multiple-kernel learning problem. MKL is able to discover discriminative features for vision tasks automatically. The underlying idea for employing the MKL approach is that a certain action class is usually only associated with a subset of kinematic joints of the articulated human body. In our case, 3D actions are represented as a linear combination of

joints, where each joint is associated with a weight. This weighted joint model is more robust to noisy features, and it can better characterize intra-class variations. In addition, we integrate ensemble learning with discriminative MKL classifiers. Training and combining multiple classifiers, ensemble methods [153] are state-of-the-art techniques with strong generalization abilities.

Our contribution is therefore an original approach by combining unsupervised feature learning and discriminative feature mining to recognize 3D human actions. In summary, the novelty of our approach is four-fold. 1) Our algorithm is unsupervised and rather generic, and may therefore be applicable to a wider range of problems with unlabeled sensor data. To the best of our knowledge, this approach is the first attempt to learn spatio-temporal features from depth video data and skeleton data in an unsupervised way. 2) We propose an ensemble learning approach with discriminative multi-kernel learning classifiers, which allows for a better characterization of inter-class variations in the presence of noisy or erroneous skeleton data. 3) We improve our performance in terms of the recognition accuracy on MSRAction3D dataset [154] (see Table 4.4), and show an accuracy superior to the state-of-the-art. 4) We further investigate our model and analyze the efficiency of a 3D action recognition system. We find that a small subset of joints ($1-6$ joints) is sufficient to perform action recognition if action classes are targeted. This observation is important to allow online decisions and improvements to the efficiency of action recognition tasks.

The remainder of this chapter is organized as follows: Section 4.1 gives details of learning the 3.5D Representation for depth video data. Section 4.2 presents the ensemble learning approach with discriminative MKL classifiers. Section 4.3 provides the evaluation results as well as a detailed analysis of our approach. Section 4.4 discussed the key contributions of our method.

4.1 3.5D Depth Video Representation

We employ the unsupervised feature learning framework in Chapter 3 for the low-level spatio-temporal feature extractions. We develop a new representation of human action, namely *3.5D Depth Video Representation*, based on the LISA and GISA features. It corresponds to the outcome of reconstructing 3.5D information from spatio-temporal features (LISA and GISA features) and the skeleton data (3D joint positions).

Compared to interest point-based methods, which describe parts of interest in the scene, LISA and GISA features are used to describe general parts of the scene as they do not need

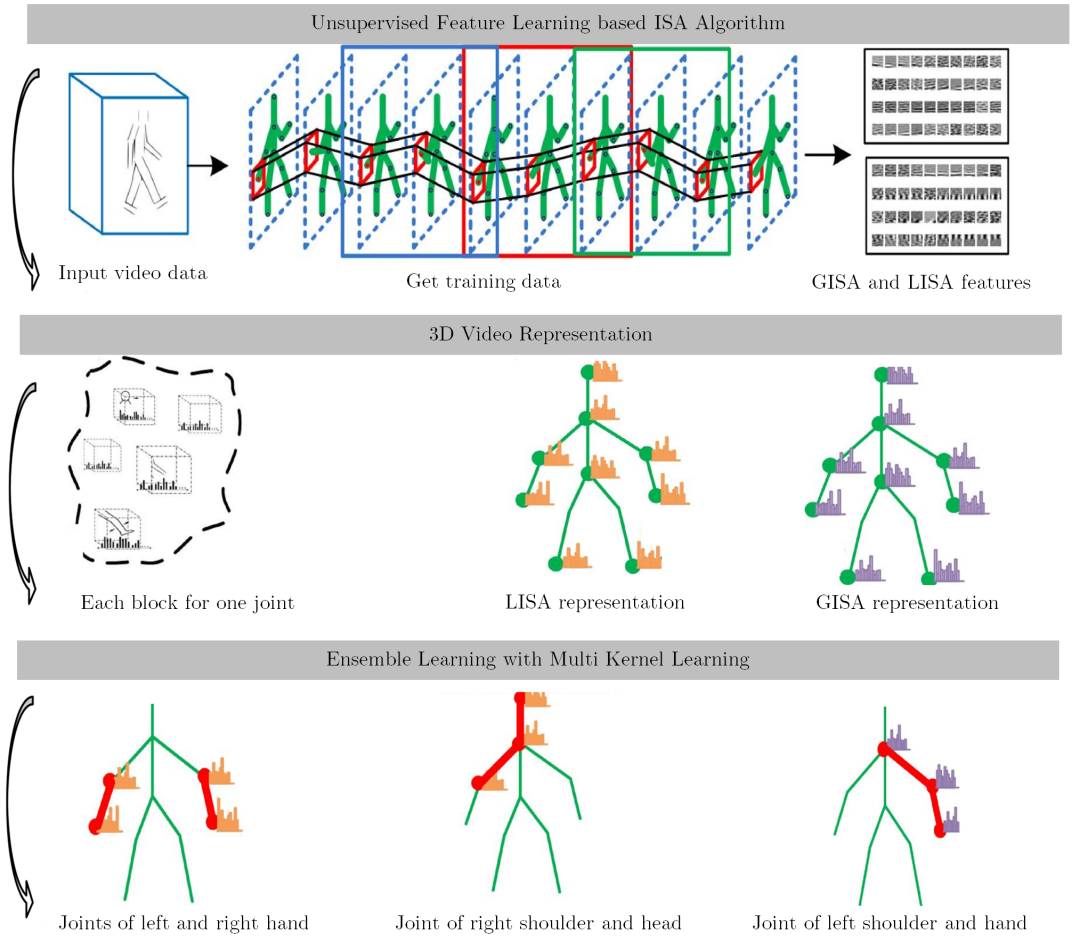


Figure 4.1: The general framework of our proposed approach. Our framework consists of two main parts: unsupervised feature learning and discriminative feature mining. We develop two types of spatio-temporal features from depth video data using independent subspace analysis, and apply an ensemble approach with discriminative multi-kernel-learning classifiers (best viewed in color).

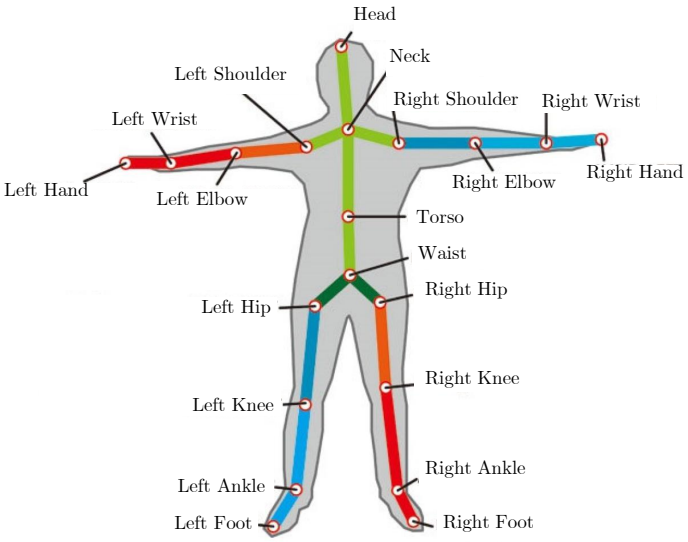


Figure 4.2: Naming of the human joints tracked by the skeleton tracker.

apply interest point detection. This approach may save the time of interest point detection process, however it can be potentially time-consuming when the dimension of the input data is large. It is generally agreed that knowing the 3D joint position of human subject is helpful for action recognition. We therefore develop a 3.5D depth video representation to combine the 3D configuration of human skeletons and spatio-temporal features of each joint. In our implementation, we utilize LISA and GISA features as spatio-temporal features.

We borrow the term, *3.5D representation*, from stereoscopic vision [155], in which they use a 2.5 representation to describe actions in static imagery. A 3.5D representation $\mathcal{G}^{\mathcal{X}}$ describing a depth video $\mathcal{X}$ consists of V nodes connected by E edges. The nodes correspond to a set of key points (joints) of the human body. A node v is represented by the 3D position of this node p_v and the histogram feature $f_v^{\mathcal{X}}$ extracted in a image region surrounding this node in time. Adjacent nodes v and v' are connected by edge e. Finally, the 3.5D representation of a depth video is written as $\mathcal{G}^{\mathcal{X}} = \{f_{v_1}^{\mathcal{X}}, f_{v_2}^{\mathcal{X}} ... f_{v_k}^{\mathcal{X}}\}$, where k denotes the number of the joints.

For a human subject in a depth video $\mathcal{X}$, the skeleton tracker tracks 20 joint positions [5] (see Fig. 4.2), which correspond to 20 nodes of a 3.5D representation $\mathcal{G}^{\mathcal{X}}$. For each joint i at frame t, its surrounding region S_t^i is of size (v_x, v_y) pixels. Let $\mathcal{T}$ denote the temporal dimension of the depth video $\mathcal{X}$. The depth video $\mathcal{X}$ is represented as the set of joint volumes $\{JV_1, JV_2 ... JV_{20}\}$. Each joint volume can be considered as a sequence of depth image patches $JV_i = \{S_1^i, S_2^i ... S_t^i\}$. The size of JV_i is $v_x \times v_y \times \mathcal{T}$ (see Fig. 4.3). Finally, $\mathcal{G}^{\mathcal{X}}$ is rewritten as

Table 4.1: Implementation details of six types of histogram features used in 3.5D depth video representations

Video Data	Spatio-temporal Feature	Joint/Joint Pair	Histogram Operation	Histogram Feature
Depth + Skeleton	GISA feature	Joint	N/A	Joint_GISA
		Joint Pair		Joint_GISAp
	LISA feature	Joint		Joint_LISA
		Joint Pair		Joint_LISAp
	GISA+LISA feature	Joint	Concatenation	Joint_GL_ISA
		Joint Pair		Joint_GL_ISAp

$\mathcal{G}^{\mathcal{X}} = \{\mathcal{F}(JV_1), \mathcal{F}(JV_2)...\mathcal{F}(JV_k)\}$, where $k = 1, ..., 20$; $f^{\mathcal{X}}_{v_k} = \mathcal{F}(JV_k)$; $F(\cdot)$ is a function where the input joint volume JV_k is converted into a histogram feature $f^{\mathcal{X}}_{v_k}$.

As the surrounding region of each joint is small compared to the whole image, we reduce the dimensionality and greatly improve efficiency. Additionally, it is possible to dense sample the local region of a joint to capture more discriminative information. Moreover, the features are discriminative enough to characterize variations in different joints. Based on the above stacked ISA model, we compute the spatio-temporal features directly from JV_i for each joint. We treat the spatio-temporal features as the resulting descriptors of the spatio-temporal interest points. Each interest point is represented by a subvolume, which consists of s_t depth image patches of size $s_x \times s_y$ (see Fig. 4.3). The spatio-temporal interest points of GISA features and LISA features are *global depth subvolumes* and *local depth subvolumes*, respectively. We dense sample the interest points from JV_i. Then, we perform a vector quantization by clustering the spatio-temporal feature from the 20 joints, which result in a bag-of-word histogram feature of each joint. With two types of spatio-temporal features (LISA and GISA features), we obtain two histogram features at each joint, named $Joint\_GISA_i$, $Joint\_LISA_i$ (see Fig. 4.3 and Table 4.1). For each joint, we apply histogram operations (e.g. concatenation) to the histograms $Joint\_GISA_i$, $Joint\_LISA_i$, which results in a new histogram feature $Joint\_GL\_ISA_i = [Joint\_GISA_i, Joint\_LISA_i]$. The concatenation operation fuses two types of histogram features to provide robustness to classification problems, a technique which has proven useful in the image classification domain [156]. As different features present human actions from different perspectives, concatenation can further be enhanced by introducing broader characteristics. For this, each 3D joint is associated with two histogram features $Joint\_GISA_i$, $Joint\_LISA_i$ and their concatenation $Joint\_GL\_ISA_i$. Each of these correspond to the feature $f^{\mathcal{X}}_v$ of a node v in $\mathcal{G}^{\mathcal{X}}$. In the following, we will refer to these three as *joint-based* features.

Inspired by the Spatial Pyramid approach [156], we group adjacent joints together as a

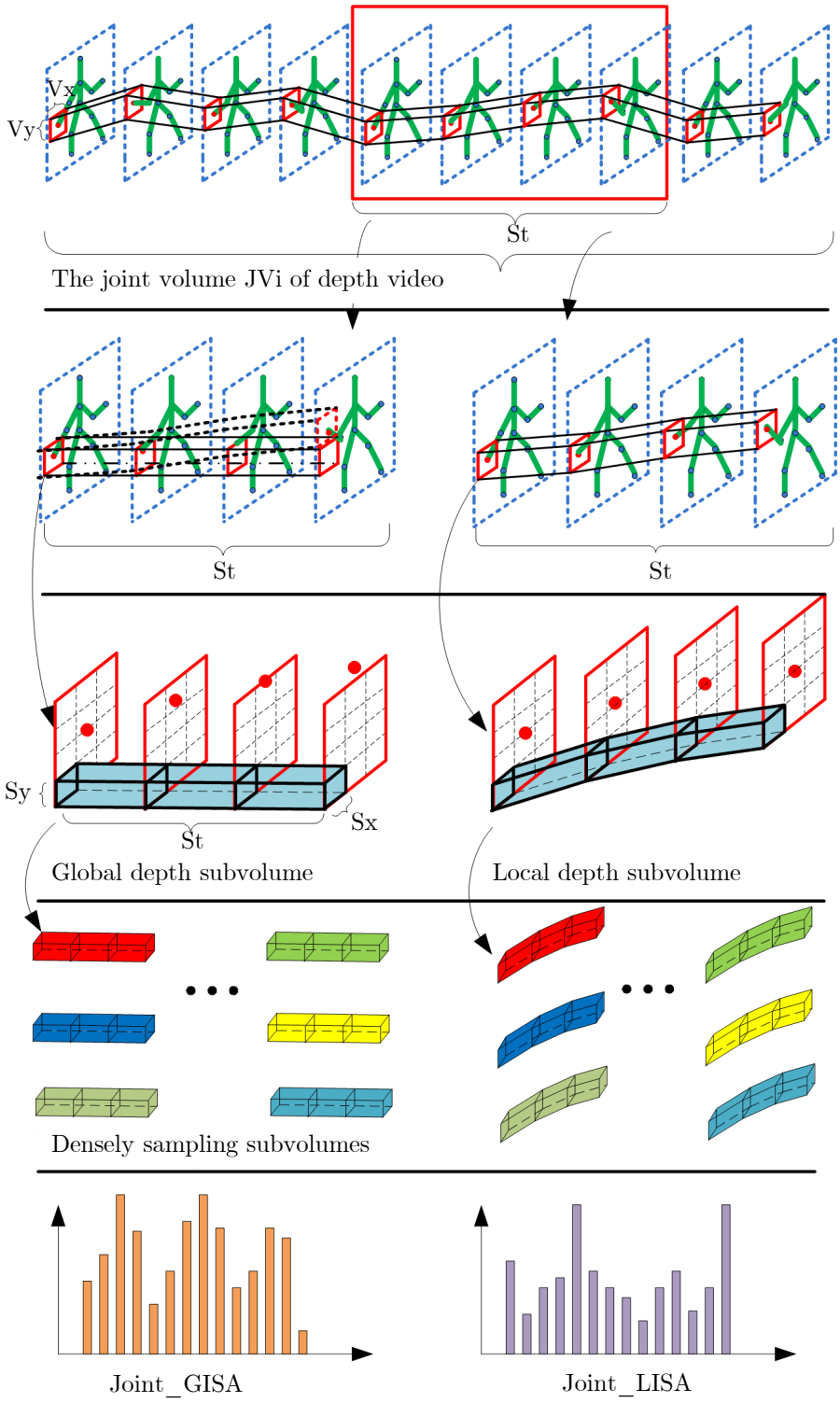


Figure 4.3: The processing steps of learning Joint_GISA features and Joint_LISA features (best viewed in color).

joint pair, seeking to capture the hierarchical structure of the skeleton. In our human skeleton model, there are 19 *joint pairs*. Each *joint pair* is represented by three histogram features $Joint\_GISAp_{ij}$=[$Joint\_GISA_i$, $Joint\_GISA_j$], $Joint\_LISAp_{ij} = [Joint\_LISA_i, Joint\_LISA_j]$, and their concatenation $Joint\_GL\_ISAp_{ij} = [Joint\_GL\_ISAp_i,\ Joint\_GL\_ISAp_j]$. We call them *joint-pair-based features*. In total, we have defined six types of 3.5D representations $\mathcal{G}^{\mathcal{X}}$ for depth video $\mathcal{X}$ (see Table 4.1). For each type of representation, the features $f_v^{\mathcal{X}}$ are given by $Joint\_GISA_i$, $Joint\_LISA_i$, $Joint\_GL\_ISA_i$, $Joint\_GISAp_{ij}$, $Joint\_LISAp_{ij}$, and $Joint\_GL\_ISAp_{ij}$, respectively. To clarify further explanations, we will omit the subscript from the above six features in the rest of this chapter.

4.2 Ensemble Learning with Discriminative MKL Classifiers

In order to model intra-class variations better and provide more robustness against errors of the skeleton tracker [5], we propose an ensemble learning approach for action recognition using depth videos. The aim of our method is to learn a discriminative subset of joints for each action class. To achieve this, we combine two concepts: (1) *Discriminative training* to explore the 3.5D representation effectively; (2) *Ensemble learning* to learn a stronger classifier at a high efficiency. Specifically, we develop an ensemble multi-kernel learning framework (EnMkl) where each component classifier is a discriminative MKL classifier that is trained on a subset of training samples. In our setting, the discriminative training and ensemble learning can benefit from each other. The MKL framework allows us to consider a subset of joints at a time, which allows us to explore the 3.5 representation efficiently, and in a systematic way. Ensemble learning selects a subset of training samples to explore the diversity of the sample data and therefore it can balance the distribution of the dataset (especially for a small size dataset) and reduce redundancy in the feature set.

An overview of the EnMkl approach we use is shown in Algorithm 1. We first describe the framework of our algorithm. Next, we give more details of the kernel design of the component classifiers.

4.2.1 Multi-kernel Learning

Joint-based features provide useful, characteristic data to allow action recognition. However, redundant or irrelevant information may complicate classification; typically, joint data may be

Algorithm 1 EnMkl

Input:

For the training set of each action class, select all positive samples $\mathcal{P}$, and all negative samples $\mathcal{N}$, $|\mathcal{P}| < |\mathcal{N}|$, $y^i \in \{+1, -1\}$ are their class labels. Define T as the number of iterations to train an AdaBoost ensemble $\mathcal{C}$.

Weights initialization for each sample: $r_\tau^i = 1/(|\mathcal{P}| + |\mathcal{N}|)$, $i = 1, ..., |\mathcal{P}| + |\mathcal{N}|, \tau = 1$, $mode = top$

while $\tau \leq T$ **do**

Weights normalization:

$$\bar{r}_\tau^i = \frac{r_\tau^i}{\sum_i r_\tau^j}, \qquad \forall i \tag{4.1}$$

if $mode == top$ **then**

Select top weighted samples: a subset $\mathcal{N}_\tau$ from $\mathcal{N}$

end if

Train an MKLSVM component classifier, $\mathcal{F}_\tau$ on $\mathcal{P}$ and $\mathcal{N}_\tau$

Compute the performance of $\mathcal{F}_\tau$ over $\mathcal{P}$ and $\mathcal{N}$:

$$p_\tau = \sum\nolimits_i r_\tau^i g_\tau^i (1 - abs(sgn(\mathcal{F}_\tau^i) - y^i)) \tag{4.2}$$

where $g_\tau^i = ((1 - sgn(\mathcal{F}_\tau^i))/2 + \ pro(\mathcal{F}_\tau^i) sgn(\mathcal{F}_\tau^i))$

pro() denotes the probability output of $\mathcal{F}_\tau^i$

Choose $\alpha_\tau = -\frac{1}{2} log(\frac{1-p_\tau}{p_\tau})$

if $\alpha_\tau > \theta$ **then**

$mode = top$; $\tau = \tau + 1$

Update the weights:

$$r_\tau^{i+1} = \bar{r}_\tau^i e^{(-2|g_\tau^i| + \alpha_\tau)(1 - abs(sgn(\mathcal{F}_\tau^i) - y^i))} \qquad \forall i \tag{4.3}$$

else

$mode = random$; Select a random subset $\mathcal{N}_\tau$ from $\mathcal{N}$

continue

end if

end while

Output:

$$\mathcal{C} = \frac{\sum_{\tau=1}^T \alpha_\tau pro(\mathcal{F}_\tau)}{\sum_{\tau=1}^T \alpha_\tau} \tag{4.4}$$

very noisy when occlusions occur, hindering the classifier from isolating relevant information.

When dealing specifically with skeletal data obtained by skeleton tracker [5] from an RGBD camera, it can be seen that some joints are more important than others with respect to action recognition. Therefore, taking this observation into account, we investigate discriminative joint subsets for human actions by the MKL algorithm. MKL is used to learn an optimal combination of *joint-based* (or *joint pair-based*) features $\{f^{\mathcal{X}}_{v_1}, f^{\mathcal{X}}_{v_2} ... f^{\mathcal{X}}_{v_k}\}$. With each kernel corresponding to each feature, different weights are learned for each joint. Weights can therefore highlight more discriminative joints for an action and ignore irrelevant or unnecessary joints by setting their weight to zero.

4.2.2 Ensemble Learning with MKL Classifiers

The properties of training datasets such as size, distribution and number of attributes significantly contribute to the generalization error of a learning machine. In action recognition tasks, class imbalances or unevenly distributed sample data is rather common. Because of the large effort of acquiring video data and manually annotating these data, the size of the training data for action recognition is typically smaller than in other computer vision tasks. In addition, different subjects perform actions with considerable variation. These complications may—without precautions being taken—lead to models that suffer from over-fitting.

To deal with these problems, randomization with under-sampling is an effective method. This technique uses a subset of majority class samples to train a classifier. Although the training set becomes balanced and the training process becomes faster, standard under-sampling often suffers from the loss of helpful information concealed in the ignored majority class samples. Inspired by [157], our method considers the distributions of different samples in the training dataset. Rather than randomly sampling subsets of the majority class, we try to balance randomization and discrimination during the training phase of the stronger classifier. For this, we define a threshold θ to evaluate component classifiers in the ensemble learning framework. This way, our algorithm can use random or discriminative sampling subsets of training samples to train a component classifier according to the performance of the component classifier in the previous iteration. (In a control experiment, we limit this ability by using only randomly sampling subsets, observing the recognition rate to drop by 0.7%.) Similar to other ensemble learning approaches, the AdaBoost algorithm [158] is used in our method to train a number of weighted component classifiers. An ensemble of all component classifiers together creates the final classifier. Here, for each class, a multiple kernel learning (MKL) classifier is used as the

base learner of an ensemble. MKL is able to mine the dominating sets of joints and learn a linear combination of these discriminative joint-based features, details of which we present in the following section.

4.2.3 Kernel Design of Component Classifiers

Our objective is to learn a component classifier that, rather than using pre-specified kernels, use kernels that are linear combinations of given base kernels. Suppose that the bags of the depth video $\mathcal{X}$ are represented as:

$$f_{\mathcal{X}} = \{f_1, f_2, ..., f_{t-1}, f_t\} \tag{4.5}$$

where t is the number of the features for each depth video. The classifier defines a function $\mathcal{F}(f^{\mathcal{X}})$ that is used to rank the depth video $\mathcal{X}$ by the likelihood of containing an action of interest.

The function $\mathcal{F}$ is learned, along with the optimal combination of histogram features $f^{\mathcal{X}}$, by using the Multiple Kernel Learning techniques proposed in [159]. The function $\mathcal{F}(f^{\mathcal{X}})$ is the discriminant function of a Support Vector Machine, and is expressed as

$$\mathcal{F}(f^{\mathcal{X}}) = \sum_{i=1}^{M} y_i \alpha_i K(f^x, f^i) + b \tag{4.6}$$

Here, f^i, $i = 1, ..., M$ denote the feature histograms of M training depth video data sets, which are selected as representative by the SVM. $y^i \in \{+1, -1\}$ are their class labels, and K is a positive definite kernel, obtained as a linear combination of base kernels

$$K(f^{\mathcal{X}}, f^i) = \sum_j w_j K(f_j^{\mathcal{X}}, f_j^i) \tag{4.7}$$

MKL learns both the coefficient α_i and the kernel combination weights w_j. For a multi-class problem, a different set of weights $\{w_j\}$ is learned for each class. We choose a one-vs.-rest strategy to decompose the multi-class problems.

Because of linearity, Eq. 4.6 can be rewritten as

$$\mathcal{F}(f^{\mathcal{X}}) = \sum_j w_j \mathcal{F}(f_j^{\mathcal{X}}) \tag{4.8}$$

where

$$\mathcal{F}(f_j^{\mathcal{X}}) = \sum_{i=1}^{M} y_i \alpha_i K(f_j^x, f_j^i) + b \tag{4.9}$$

Table 4.2: Partitioning of the MSRAction3D dataset into three subsets as used in our evaluation

Cross Subset 1 (CS1)	Cross Subset 2 (CS2)	Cross Subset 3 (CS3)
Tennis Serve (TSr)	High Wave (HiW)	High Throw (HT)
Horizontal Wave (HoW)	Hand Catch (HC)	Forward Kick (FK)
Forward Punch (FP)	Draw X (DX)	Side Kick (SK)
High Throw (HT)	Draw Tick (DT)	Jogging (JG)
Hand Cap (HCp)	Draw Circle (DC)	Tennis Swing (TSw)
Bend (BD)	Hands Wave (HW)	Tennis Serve (TSr)
Hammer (HM)	Forward Kick (FK)	Golf Swing (GS)
Pickup Throw (PT)	Side Boxing (SB)	Pickup Throw (PT)

With each kernel corresponding to each feature, there are 20 weights w_j to be learned for the linear combination of the *joint-based* features, and 19 weights w_j to be learned for the *joint pair-based* features. These weights represent how discriminative a joint is for an action; we can even ignore less discriminative joints by setting w_j to zero.

As MKL cannot give a posterior class probability $P(y = 1|X)$, we propose an approximation of the posteriors by a sigmoid function

$$P_m(y = 1|\mathcal{X}) \approx pro(\mathcal{F}_\tau^{\mathcal{X}}) \equiv \frac{1}{1 + \exp(A_m \mathcal{F}_\tau^{\mathcal{X}} + B_m)} \tag{4.10}$$

We follow Platt's method to learn A_m and B_m [160]. For each MKL model m, we then learn a sigmoid function $pro(\mathcal{F}_\tau)$.

4.3 Evaluation

In this section, we first compare our algorithm quantitatively against current state-of-the-art 3D action recognition algorithms, measuring recognition accuracies on the MSRAction3D dataset. After that, we further analyze the efficiency of our approach in a 3D action recognition system. In addition, we study the general advantages of discriminative MKL classifiers in the field of action recognition. In a more specific evaluation, we discuss the discriminative joint subset for each action class, and we study how many joints in a depth video are sufficient to perform certain action detection and recognition tasks in our framework.

Table 4.3: Recognition accuracy of our method on each of the three subsets. CS1, CS2 CS3 are the abbreviations of Cross Subset 1, Cross Subset 2, Cross Subset3 (see Table 4.2).

Method	CS1	CS2	CS3
EnMkl-s +$Joint\_GL\_ISAp$	**0.882**	0.898	0.951
EnMkl + $Joint\_GL\_ISAp$	0.881	**0.927**	**0.959**

4.3.1 Experimental Setup

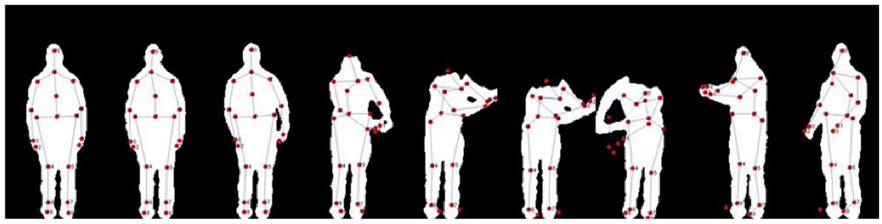
The MSRAction3D dataset [154] is a public dataset that provides sequences of depth maps and skeletons captured by a Kinect RGBD camera. It includes 20 actions performed by 10 subjects facing the camera during performance. Each subject performs each action two or three times. As shown in Fig. 4.4, actions in this dataset reasonably capture a variety of motions related to arms, legs, torso, and their combinations. In order to facilitate a fair comparison, we follow the same experimental settings as [161, 162, 26] to split 20 actions into three subsets as listed in Table 4.2, each having 8 action classes. In each subset, half of the subjects are selected as training data and the other half for testing; we perform a two-fold cross validation.

4.3.2 Sensitively Analysis

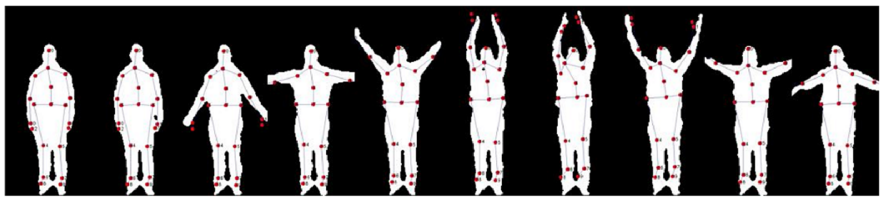
We analysis the effect of several parameters of our model: the size of the input unit of ISA model, dense sampling stride, codebook size, kernel type. We show the results across different parameter setting for LISA and GISA features by using a three-fold cross-validation on training data: Cross Subset 1 (See Table 4.3).

We first evaluate the effect of the size of the input units. The input units to the bottom layer of ISA model are of size $s_x \times s_y \times s_t$. We report results of our model with different spatial size s_x ($s_x = s_y$) and different temporal size s_t of the input units. Fig. 4.5 shows the average classification accuracies using cross-validation. Increasing the spatial and temporal size of the input units improves the performance up to $s_x = 12$. This is probably due to the fact that input units need to have a minimum size to dense sample enough interest points. We observe the best result with the size of the input unit of 12 pixels $\times$12 pixels $\times$10 frames.

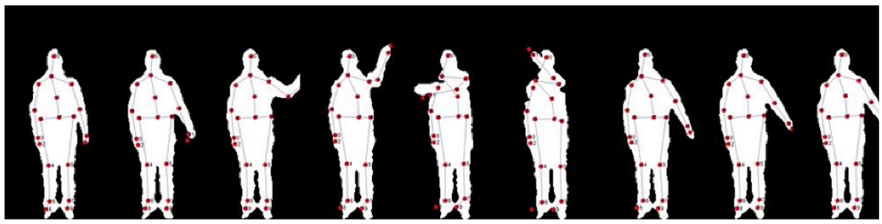
With respect to the dense sampling stride, Fig. 4.6 presents the results for 1 pixel to 4 pixels. The performance increases with a higher sampling density. This is also consistent with dense sampling at regular position where more features in general improve the results [6]. A



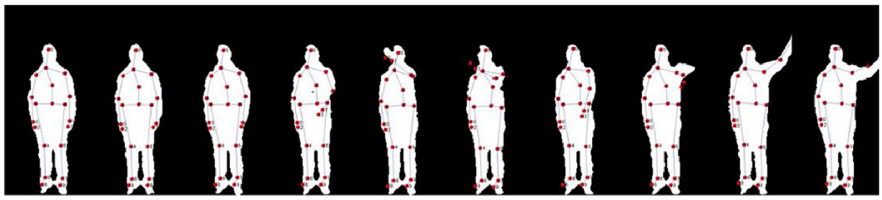
(a) Golf Swing



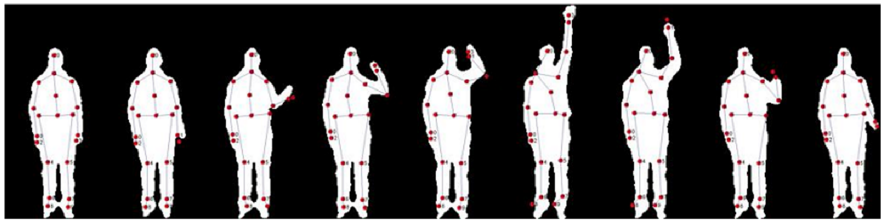
(b) Hand Clap



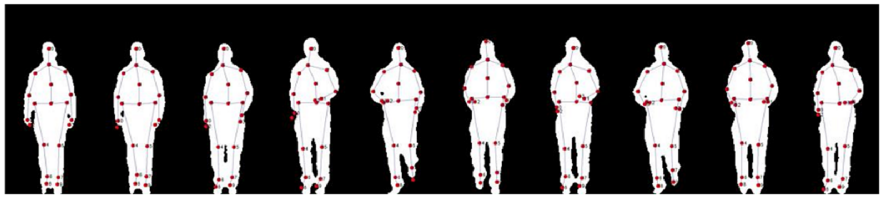
(c) Draw X



(d) Draw Tick

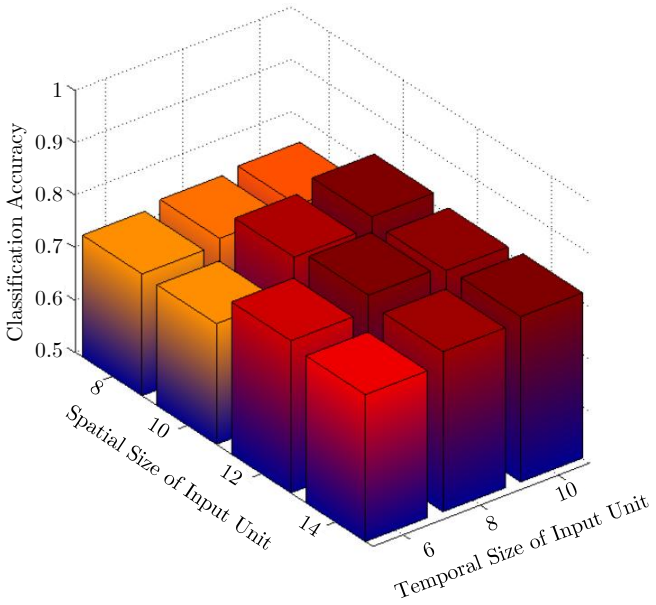


(e) High Throw

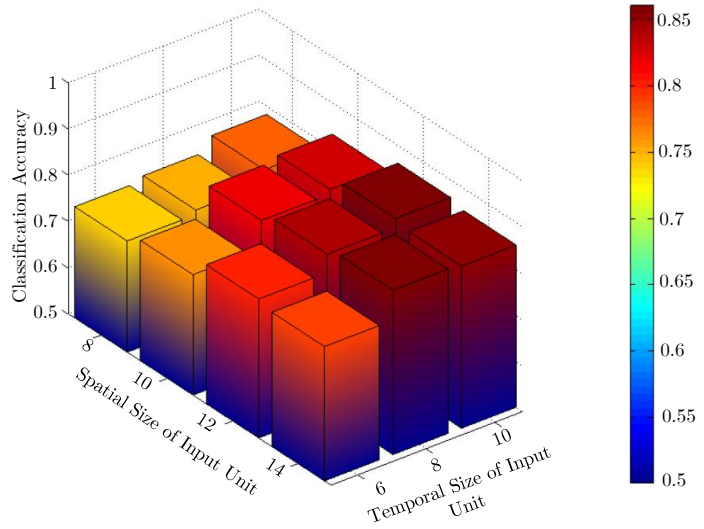


(f) Jogging

Figure 4.4: The sequences of depth maps and skeleton for different action classes. Each depth image includes 20 joints (marked as red points).



(a) Classficiation results with GISA features



(b) Classification results with LISA features

Figure 4.5: Effect of spatial and temporal size of the input units with GISA feature (a) and LISA feature (b) on classification accuracy using cross-validation.

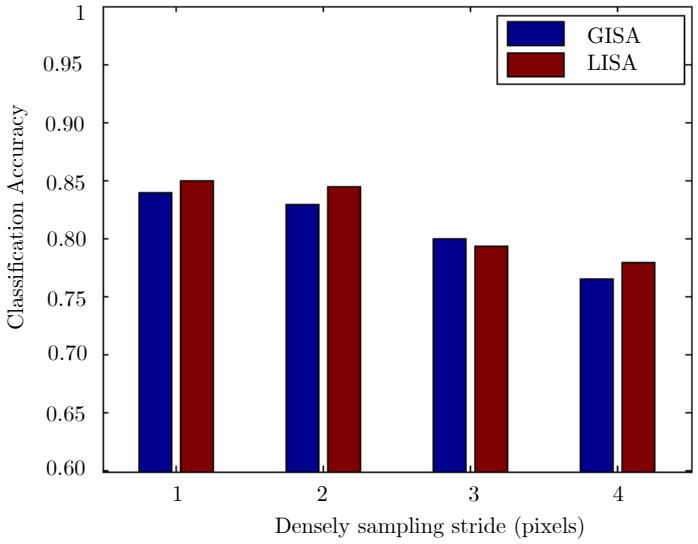


Figure 4.6: Effect of the dense sampling stride with GISA feature and LISA feature on classification accuracy using cross-validation.

sampling stride of 1 pixels samples every pixel which increase the computational complexity. We set the dense sampling stride as 2 pixels offers a good trade-off between speed and accuracy.

Fig. 4.7 shows the classification performance for different combinations of kernels and codebook sizes. The χ^2 kernel outperform the intersection kernel. Larger codebook sizes have been reported to improve the classification performance. For both kernels, the performance saturates at codebook size = 700 or codebook size = 900.

4.3.3 Model Details

We use the found optimal parameters of sensitively analysis to train our models and test our method. We train the ISA model on the MSRAction3D training sets. The input units to the bottom layer of ISA model are of size $12 \times 12 \times 10$, which are the dimensions of the spatial and temporal size of the subvolumes. The size is the same for global depth subvolumes (for learning GISA features) and local depth subvolumes (for learning LISA features). The subvolumes of the top layer of the ISA model are of the same size as those of the bottom layer.

We perform vector quantization by k-means on the learned spatio-temporal features for each joint. For the distance parameter in the dense sampling step for the local regions of each

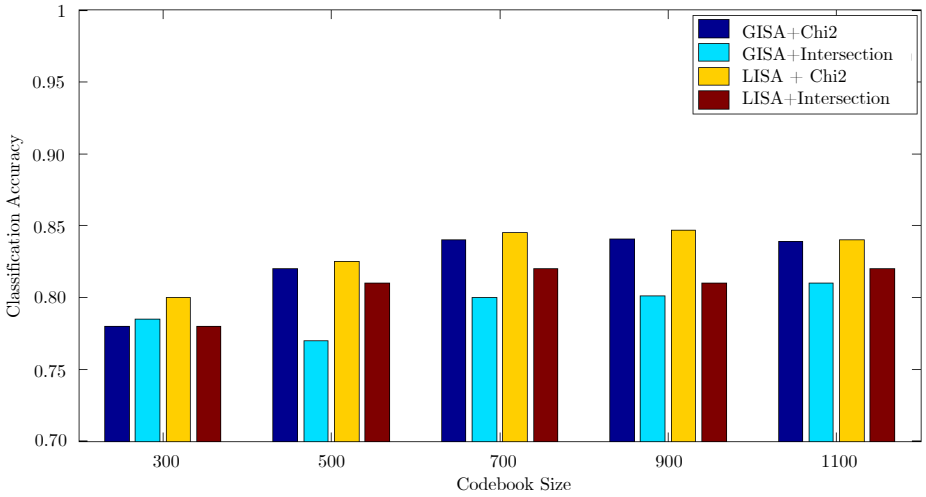


Figure 4.7: Effect of the codebook size and kernel type with GISA feature and LISA feature on classification accuracy using cross-validation.

joint, we choose a region of 2 pixels. The codebook size k is 700 for both $Joint\_GISA$ feature and $Joint\_LISA$ feature. Therefore, each depth video is represented by 20 histogram features for $Joint\_GISA$, $Joint\_LISA$, $Joint\_GL\_ISA$ or 19 histogram features for $Joint\_GISAp$, $Joint\_LISAp$, and $Joint\_GL\_ISAp$. We choose χ^2 as the histogram kernel for the multi-class SVM classifier. For EnMkl, we set the number of subsets $|\mathcal{N}_\tau| = 3|\mathcal{P}|$, and the rounds of the AdaBoost $T = 20$. The threshold for a good component classifier is set to 1.45. Across the three subsets, all parameters are set to the same values. Note that when we set the number of the samples in subsets $|\mathcal{N}_\tau| = |\mathcal{N}|$, and the rounds of the AdaBoost $T = 1$, EnMkl becomes equivalent to a multi-kernel learning problem; we call this special case EnMkl-s.

4.3.4 Experimental Results

We compare our algorithm with several recent methods including: (1) Li et al. [161], where bags of 3D points are sampled from depth maps and an action graph is employed to model the dynamics of the actions; (2) Yang and Tian [38], who design a new type of feature set based on position differences of joints; (3) Wang et al. [106], where the depth sequence is randomly sampled and the most discriminative samples are selected and described using LOP descriptors; (4) Wang et al. [20], where local occupancy pattern features are used over the skeleton joints; (5) Oreifej and Liu [162], who describe the depth sequence using a histogram that captures the distribution of surface normal orientations in 4D space; (6) Xia and Aggarwal [26], who employ a filtering method to extract STIPs from depth videos; (7) Wang et al. [141], where observations are represented by histograms of activating spatial and temporal part-sets; (8) Gowayyed et al. [144], who design a 2D trajectory descriptor, the histogram of oriented displacements (HOD).

A comparison of our method against the best published results for the MSRAction3D dataset is reported in Table 4.4. Because we test six types of 3.5D representations $\mathcal{G}^{\mathcal{X}}$ for two models EnMkl and EnMkl-s, Table 4.4 shows 12 results in total. As can be seen from the table, our approach outperforms a wide range of methods. There is a clear increase in performance of our method EnMkl (with $Joint\_GL\_ISAp$ feature) (92.3%) compared to the closest competitive method (91.3%). Note that the absolute performance is very good, considering that failures in the skeleton tracker are quite frequent and tracked joint positions are rather noisy. The obtained accuracy of EnMkl-s (with $Joint\_GL\_ISAp$ features), a special case of EnMkl without using ensembles, is 91.2%. These encouraging results illustrate the effectiveness of our unsupervised learning features.

Table 4.4: Comparison of recognition accuracy between previous methods and our proposed approach on MSRAction3D dataset

Method	Accuracy
Action Graph On Bag of 3D Points [154]	0.747
EigenJoints [38]	0.823
Random Occupancy Pattern [106]	0.865
Mining Actionlet Ensemble [20]	0.882
Histogram of Oriented 4D Normals [162]	0.889
Spatio-Temporal Depth Cuboid Similarity Feature [26]	0.893
Pose-based Action Recognition [141]	0.902
Histogram of Oriented Displacements [144]	0.913
EnMkl-s + $Joint\_GISA$	0.879
EnMkl-s + $Joint\_GISAp$	0.896
EnMkl-s + $Joint\_LISA$	0.895
EnMkl-s + $Joint\_LISAp$	0.912
EnMkl-s + $Joint\_GL\_ISA$	0.894
EnMkl-s + $Joint\_GL\_ISAp$	**0.914**
EnMkl + $Joint\_GISA$	0.887
EnMkl + $Joint\_GISAp$	0.901
EnMkl + $Joint\_LISA$	0.903
EnMkl + $Joint\_LISAp$	0.920
EnMkl + $Joint\_GL\_ISA$	0.903
EnMkl + $Joint\_GL\_ISAp$	**0.923**

Compared to EnMkl-s, the improvement of EnMkl is about 1%. This indicates that the ensemble learning approach can better capture intra-class variations and is more robust against noise and errors in depth maps and joint positions. This observation is consistent with [20], who report that accuracy decreases when the ensemble approach is disabled in their experiments. It is also important to note that in our methods, accuracies obtained using LISA features (91.2% for EnMkl-s with $Joint\_LISAp$) are better than using GISA features (89.6% for EnMkl-s with $Joint\_GISAp$). This is probably because the skeletons have a natural correspondence over time, and LISA features can model spatial structures more explicitly than GISA features. To further investigate the relationship between LISA features and GISA features, we study the most important joints discovered by $Joint\_LISA$ and $Joint\_GISA$ features with EnMkl-s method. For each action class, the *top-weighted* joint is selected as the most important joint. Here, we define the *top-weighted* joint as the joint with the highest maximum. With $Joint\_LISA$ features, *right hand*, *right wrist*, and *left wrist* joints (the top three) receive the most votes in 20 action classes in MSRAction3D dataset. With $Joint\_GISA$ features, *right hand*, *right shoulder*, and *left elbow* joints receive the most votes (the top three). The results indicate that LISA and GISA features have some qualities in common, as both of them select *right hand* as the highest weighted joint. Our results are consistent with [144], who conducted an experiment using features from only one joint to perform action recognition. Their results show that using the right hand joint outperforms all other joints on MSRAction3D dataset. Additionally, it is interesting to note that in our methods, accuracy obtained using $Joint\_GISAp/Joint\_LISAp$ features is 90.1%/92% (EnMkl), which is better than using $Joint\_GISA/Joint\_LISA$ features 88.7%/90.3% (EnMkl). These results show a clear advantage of the spatial pyramid approach, even though we simply group adjacent joints together as *joint pairs* and capture the hierarchical structure of human skeleton.

In Table 4.3, we report average accuracies of all three test sets (*Cross Subset 1 (CS1), Cross Subset 2 (CS2), Cross Subset 3 (CS3)*), and show the best performance results of the two methods, EnMkl-s and EnMkl. In Fig. 4.8, we illustrate the average accuracies of all action class. While the performance in CS2 and CS3 is promising, the accuracy in CS1 is relatively low. This is probably because actions in CS1 are performed with rather similar movements. For example, in CS1 *Hammer* tends to be confused with *Forward Punch* and *Horizontal Wave*, and *Pickup Throw* consists of *Bend* and *High Throw*. Although our method reaches an accuracy of 100% in 12 out of 20 actions, the accuracy of the *Hammer* in CS1 is only 37.7%. This is

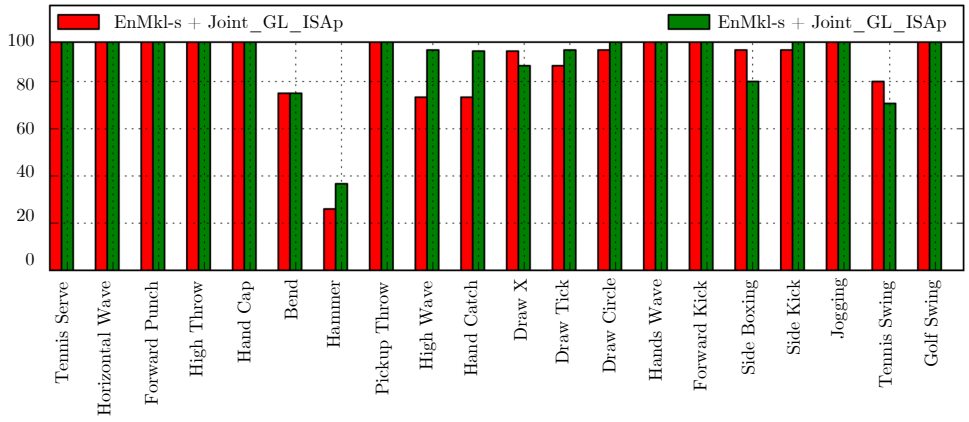


Figure 4.8: Recognition accuracies for the 20 action classes of the MSRAction3D dataset. We compare EnMkl to EnMkl-s using Joint_GL_ISAp features. All abbreviations of action classes are defined in Table 4.2 (best viewed in color).

probably due to the significant variations of the action *Hammer* performed by different subjects; recognition performance could be improved by adding more subjects to the training set.

4.4 Discussion

We present a novel ensemble learning approach, named EnMkl, which combines unsupervised feature learning and discriminative feature mining. For this, we develop two types of spatio-temporal features, applying independent subspace analysis to depth video data. Our approach is rather generic and unsupervised, and may therefore be applied to a wider range of problems with unlabeled sensor data. To the best of our knowledge, EnMkl is the first attempt to learn the spatio-temporal features from depth video data in an unsupervised way. Furthermore, we propose an ensemble learning approach with discriminative multi-kernel-learning classifiers, which allows for a better characterization of inter-class variations in the presence of noisy or erroneous skeleton data. In our evaluation, we analyze the efficiency of our 3D action recognition approach. In more detailed discussions (at the bottom), we investigate which joint subsets are discriminative for different types of actions, and we study which of these joints are sufficient to recognize these actions. Our experimental results of the EnMkl approach show a performance superior to existing techniques. Results also suggest that learning spatio-temporal features directly from depth video data may be a promising direction for future research, as combining these features with ensemble learning may further increase performance.

Advantages of Multi-kernel Learning It is generally agreed that, although the human body has a large number of kinematic joints, a certain action is usually only associated with a subset of them. Additionally, feature extraction in action recognition is usually computationally expensive. A reduced feature subset leads to lower computational costs. This encourages us to investigate the following two questions: Do more joints allow for better for action recognition? Do joints contribute equally to recognizing an action?

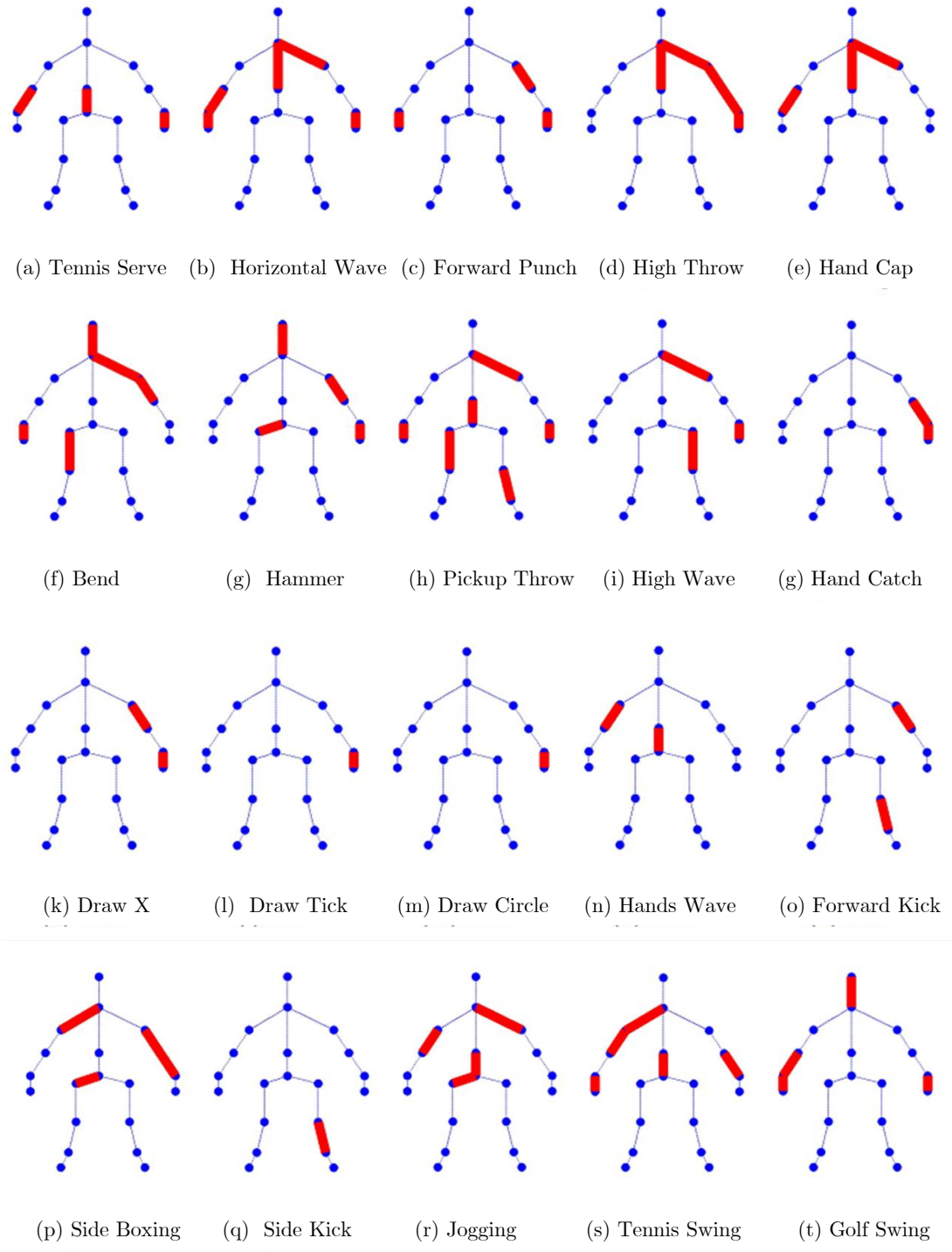
We address the first question by setting the following control experiment: we conduct two tests, where the first test uses 20 joint features with equal weights for action recognition and the second test uses a subset (the subset is obtained by EnMkl method) of joint features with equal weights (manually setting w_j to 1). We perform both tests on MSRAction 3D dataset. It is not surprising that the first test performs worse than the second, with a decline of 4.5% in accuracy on the MSRAction 3D dataset. This indicates that a subset of characteristic data may lead to a more successful recognition and a full set of data with irrelevant information may complicate the classification.

To answer the second question whether joints contribute equally to an action, we re-run the experiment with the same settings as in Section 4.3.3 and manually set w_j to 1. The results of this test show substantially worse accuracies than those of the previous experiments. More precisely, setting the weight to 1, accuracy drops by a significant amount of 5% for the MSRAction3D dataset. This confirms that the weights learned from MKL are indeed very relevant for successful action recognition.

Mining Discriminative Joint Subsets In our EnMkl-s method, each action is represented as a linear combination of joint-based features. We learn their weights in a multiple kernel learning method to obtain discriminative joint subsets.

Fig. 4.9 illustrates the skeleton with joints weights obtained by our EnMkl-s method. Here, we only show the results of the $Joint\_LISA$ features as an example; the other five feature sets would show to very similar results. The $Joint\_LISA$ features with $weights > 0$ are marked as thick, red lines. The average number of $Joint\_LISA$ features for 20 actions in the MSRAction3D dataset is four. Three of 20 action classes have only one discriminative $Joint\_LISA$ feature. This result is rather interesting: Imagining we want to recognize or detect a specific action class; we only need to extract features from one joint rather than the entire video data. This can be implemented and executed at a high efficiency.

EnwMi-s is also able to deal with tracking errors in the skeleton data and can better capture intra-class variation. Fig. 4.9(t) shows that *Golf Swing* is represented by the combination of



(a) Tennis Serve (b) Horizontal Wave (c) Forward Punch (d) High Throw (e) Hand Cap

(f) Bend (g) Hammer (h) Pickup Throw (i) High Wave (g) Hand Catch

(k) Draw X (l) Draw Tick (m) Draw Circle (n) Hands Wave (o) Forward Kick

(p) Side Boxing (q) Side Kick (r) Jogging (s) Tennis Swing (t) Golf Swing

Figure 4.9: The joint subsets used to recognize the 20 action classes in the MSRAction3D dataset. Our method can learn discriminative joint subsets for each action class. The weight associated with each joint describes how discriminative a joint is for that action. Joints with weights >0 are highlighted as thick, red lines (best viewed in color). All abbreviations of action classes are defined in Table 4.2.

the joints *head, neck, right hand, right wrist, left hand, left wrist and left elbow* (see Fig. 4.2 for the definition of the joint labels). Fig. 4.9(a) shows that *Tennis Serve* is represented by the combination of the joints *left elbow, left wrist, right hand, right wrist, torso, and waist*. It is obvious that different action classes have different discriminative joint subsets. Fig. 4.9(r) shows that *Jogging* is represented by the combination of the joints *left elbow, left shoulder, left hip, waist, torso, neck and right shoulder*. Normally, one would expect *Jogging* to be related to the left and right feet or ankles. However, in the MSRAction3D dataset, the tracked positions of the joints *right/left foot, and right/left ankle* are very noisy (see Fig. 4.4). Therefore, these joints are not discriminative for the action class *Jogging*, which is consistent with Fig. 4.4(f). This shows that our method is rather robust against tracking errors in the skeleton data.

Computational Complexity The training phase of neural networks (e.g. unsupervised feature learning) is usually computationally expensive and requires much tuning. The ISA algorithm, however, does not need the tweaking with the learning rate or the convergence criterion. For the training stage, the ISA algorithm takes 3-4 hours to learn the stacked ISA model using the setting in Section 4.3.3.

To analyze the computational comoplexity of the feature extraction, we extract features with dense sampling on 20 video clips from the MSRAction3D dataset. The run-time is obtained on a notebook with a 2.3 GHz double-core CPU and 8GB RAM. The implementation is in unoptimized and un-parallelized MATLAB. Feature extraction using our method runs 1 frame per second with the entire video and 6 frames per second with specific portions of the video (the surrounding regions of the joints of the subject which performs the action in the video). As the extraction time relies heavily on matrix vector products, it can be implemented and executed much more efficiently on a GPU.

Chapter 5

Multimodal Gesture Detection and Recognition With Randomization and Discrimination

In this chapter, We describe a novel human motion perception system called *multimodal gesture detection and recognition system with randomization and discrimination*. Our system is built on the unsupervised feature learning framework developed in Chapter 3. Unlike most traditional approaches that rely on the construction of complex handcrafted features, we are able to learn and extract spatio-temporal features from four modalities (grayscale, depth, gradient, surface normal) of RGB-D video data. In addition, we proposed to build the gesture recognizer under the random forest framework by combining two ideas, discrimination and randomization. The benefit is that the feature selection and classifier training are done in one step which is different with most existing approaches. State-of-the-art approach separate the feature mining and classifier training into two steps. Our approach is able to identify semantically meaningful spatio-temporal contents that closely match human intuition.

5.1 Introduction

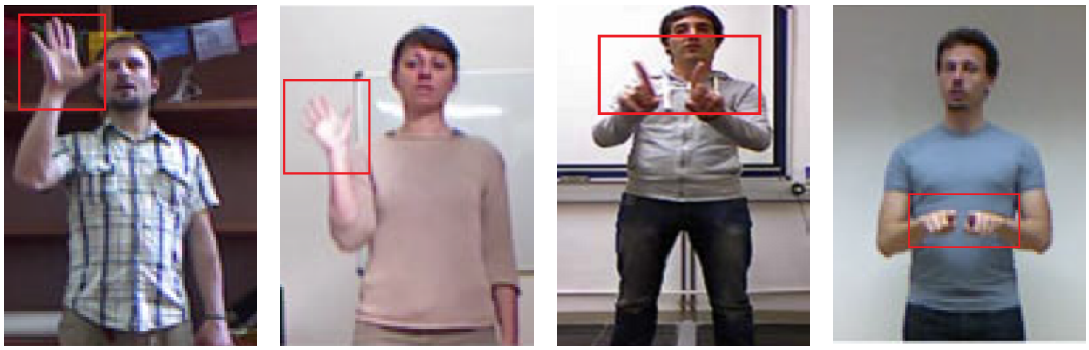
During the past decades, approaches of gesture recognition were controller-based, in which users had to wear human motion capture systems. The interfaces of users and devices are traditional command line and graphic user interfaces. With the development of input devices,

multi-modality data have become available, which directly leads to the rise of multi-modality gesture recognition. Recently, vision-based gesture recognition has become the mainstream of the research due to the good abilities that enable the controller-free and natural user interactions (NUI). Natural user interactions are based on natural interaction (e.g., gestures, speech, actions) that people use to communicate with the smart objects (e.g., computer, smartphones). Therefore, NUI have good usability and provide better user experience compared to traditional graphic user interface. Kinect, the motion sensing input device developed by Microsoft corporation, features an RGB camera, a depth sensor and a multi-array microphone. With all these features, Kinect serves as an ideal experimental platform for developing new NUI systems of multi-modality gesture detection and recognition.

Gesture detection and recognition refers to detecting and classifying meaningful motions executed by human hands. It has promising application prospects in human-computer and human-robot interaction. Compared to human actions recognition which human actions are performed by full human body, hand gestures recognition can be considered as subordinate-level categorization. Psychologists have shown that early-stage human vision system performs well on basic-level visual categorization tasks (e.g, cats vs. trees; humans vs cars) than subordinate-level visual tasks (e.g., Bulldog vs. Beagle; Exotic Shorthair vs. Persian).It is also very interesting to see that computer vision community follows a very similar path. Basic-level vision categorization task has seen great advance while subordinate-level vision task has received few studies. Therefore, developing an automated visual perception system for subordinate-level categoriztion tasks is an active demand in many applications such as human-robot and human-computer interaction system.

In this chapter, gesture recognition is regarded as a sub-ordinate level categorization problem. Unlike traditional human action recognition problems where different types of actions can be separated by human skeleton parts (e.g, hands vs. legs), more detailed and semantic visual informations need to be mined for sub-ordinate categorization. The red bounding boxs in Figure 5.1 demarcate the distinctive characteristics for different gesture classes. Here we only capture one frame of each gesture video for the demonstration. Example of human hand gestures in Figure 5.1 are coming from Chalearn Gesture dataset which will be introduced in Section 5.5.1. Figure 5.1a and Figure 5.1b show two different gesture classes *OK* and *Non ce nepiu*, which differ mainly in hand poses but not in other human body parts. Figure 5.1c and Figure 5.1d show another two different gesture classes *E un furbo* and *Buonissimo*, which differ mainly in relative position of fingers and eyes. Besides the differences coming from the

(a) OK **(b)** Non ce nepiu **(c)** E un furbo **(d)** Buonissimo



(e) Perfetto **(f)** Perfetto **(g)** Daccordo **(h)** Daccordo

Figure 5.1: Figure 5.1a and Figure 5.1b show two different gesture classes *OK* and *Non ce nepiu*, which differ mainly in hand poses but not in other human body parts. Figure 5.1c and Figure 5.1d show another two different gesture classes *E un furbo* and *Buonissimo*, which differ mainly in relative position of fingers and eyes. Besides the differences coming from the spatial space, gestures recognition has another challenges which are the gesture execution speed and scale occurring in time. Figure 5.1e and Figure 5.1f show the same gesture *Perfetto*. Figure 5.1g and Figure 5.1h show the same gesture *Daccordo*. They differ primarily in the hand movement scale (hands above the shoulder in Figure 5.1e vs. hands parallel to the shoulder in Figure 5.1f, hands in front of the heart in Figure 5.1g vs. hands in front of the shoulder in Figure 5.1h) (best viewed in color).

spatial space, gestures recognition has another challenges which are the gesture execution speed and scale occurring in time. Figure 5.1e and Figure 5.1f show the same gesture *Perfetto*, which differ primarily in the hand movement scale (hands above the shoulder vs. hands parallel to the shoulder). Methods and algorithms developed for basic-level visual categorization tasks (e.g., action recognition) are often not ready to detect such subtle differences among the sub-ordinate vision classes (e.g, hand gesture recognition). We tackle this problem from the perspective of discovering a large number of meaningfully semantic video blocks with arbitrary sizes, shapes, and locations in space and time, as well as pairs of video blocks that carry discriminative video data statistic. However, a big challenge comes from this approach: as video data is high dimensional, a short video will be able to generate billions of video blocks. Consider all the dimensions in the time and space, the amount of the candidate video blocks increase exponentially. Furthermore, these video blocks overlap significantly in the time and space. This means that many video blocks are highly correlated.

To solve these problems, we introduce the concept of *randomization* which is often used in machine learning algorithms. *Randomization* means we select a subset of video features randomly for each time. We keep the main goal *efficiency* of developing human gesture recognition system in mind, and propose a *discriminative random forest* for the gesture recognition. Unlike traditional random forest, *discriminative random forest* is built by discriminative decision trees. The purpose of using discriminative decision trees is to explore meaningful video blocks and pairs of video blocks that are highly discriminative for sub-ordinate level of gesture recognition. Our approach is able to efficiently mine a very dense sampling 3D space by using strong classifiers at each node and combining information along the different depth of the tree. Comparing to the weak classifier in classical random forest, strong classifiers significantly enhances the power of the decision trees. We integrate our gesture recognition system into a gesture detection approach. We finally develop a novel human motion perception system called *multimodal gesture detection and recognition system with randomization and discrimination.* We evaluate our system on Chalearn Gesture dataset, which is the largest gesture dataset in the literature. The primary objective of Chalearn Gesture dataset is to evaluate the performance of computational methods on gesture recognition. The emphasis is on multi-modality automatic learning of a set of 20 gestures performed by several different users, with the aim of performing user independent continuous gesture spotting. Specifically, this dataset aims at the recognition of continuous, natural human gestures with the multi-modality nature of the visual cues, as well as technical limitations such as spatial and temporal resolution and unreliable depth cues. We

show that our system achieves state-of-the-art performance on Chalearn Gesture dataset. More importantly, our approach is able to discover meaningfully semantic video blocks that performs very closely to human vision system. As semantic video blocks are intuitively discriminative information of the sub-ordinate categorization, our approach automatically discover such kind of information which indicates the efficiency of our system.

Figure 5.2 shows the architecture of the proposed multi-modality gesture detection and recognition system. The system utilize un-supervision (unsupervised learning of spatio-temporal features from four channels of RGB-D video data), randomization (exploring the spatio-temporal dense sampling space efficiently) and discrimination (discriminative training to extract the information in the video data effectively) to learn a multi-modality gesture recognizer. In Section 5.2, we describe the overall architecture of the proposed system. In Section 5.3 and 5.4, we provide the details of the individual modules that constitute our gesture recognition system. In Section 5.5, we discuss the results achieved by our system. Finally, we present a few conclusions in Section 5.6.

5.2 System Architecture

The complete architecture of the proposed multi-modality gesture detection and recognition system is shown in Fig. 5.2. It contains three main building blocks, namely unsupervised feature learning, gesture segmentation, and gesture classification. Each input sample in the Chalearn Gesture dataset contains a sequence of gestures performed by a subject and these gestures are typically separated by pauses in between. However, some of the gestures in the input sample are consecutive. Some of input samples include unrecognized gestures except for the gestures corresponding to one of the 20 gestures in the pre-defined gesture vocabulary (such as *ok* or *perfetto*). The first task of our approach is to detect the candidate gestures and temporally segment them by identifying their start and end frames. We use the skeletal joint data for gesture detection and segmentation. We assign each frame of the input sample a label: *gesture* or *non-gesture*. We extract features of each labeled frame from the skeletal joint of a 13-frame-long slide window where the labeled frame is at the center of the slide window. We train a Support Vector Machine model for each input sample. To label the frames of a test sample, our approach output the predict labels for the test sample with likelihood scores fusing the prediction confidence of the SVM models (all the models are trained on the training

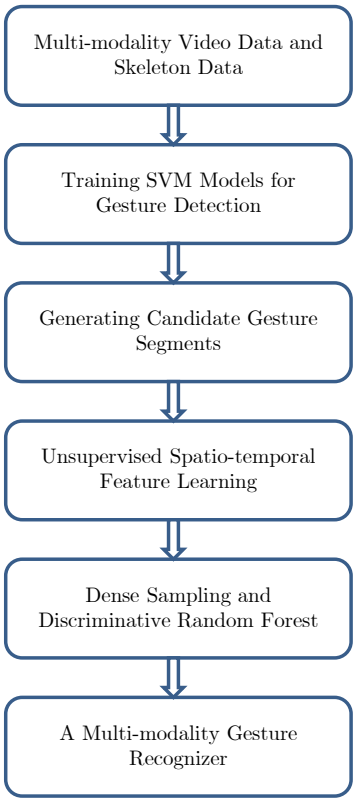


Figure 5.2: The work-flow of the proposed multi-modality gesture detection and recognition system. Our system utilize un-supervision (unsupervised learning of spatio-temporal features from four channels of RGB-D video data), randomization (exploring the spatio-temporal dense sampling space efficiently) and discrimination (discriminative training to extract the information in the video data effectively) to learn a multi-modality gesture recognizer.

samples). According with the predict labels of the test sample, we segment the sample into several candidate gestures.

Once the given input sample is broken down into candidate gesture segments, the next task is to provide a suitable representation of the candidate gesture contained within each segment. We utilize spatio-temporal features extracted from the RGB-D video data to represent the gesture. In contrast to previous work, we extracted the spatio-temporal features using an unsupervised learning approach. At the heart of unsupervised learning approach is the extension of Independent Subspace Analysis algorithm [85] for the use of RGB-D video data. To effectively model the motion patterns of the gestures for the classification, we approach this problem from the perspective of mining a large number of video blocks with arbitrary shapes, spatio-temporal

sizes, or location that carry discriminative gesture video statistics. However, this approach poses a fundamental challenge: without any feature selection, even a modestly sized video will yield millions of video blocks. In addition, as large number of the blocks overlap significantly, these blocks are highly correlated and introduce significant redundancy among these features. On the other hand, many video blocks are not discriminative for distinguishing different gesture classes. To address this issue, we propose a random forest with discriminative decision trees approach to mine video blocks that are highly discriminative for the gesture classification tasks. Unlike traditional decision trees [163], our approach uses a SVM classifier at each node and integrates information at different depths of the tree to effectively mine a very dense sampling space of the video data. The final predicted label for a candidate gesture is assigned to the class which maximum the average of the posterior probability from the leaf node of each tree.

We implement various modules in the system using off-the-shelf tools available on different platforms. For example, pre-processing of the video data were implemented on Python. Unsupervised learning of spatio-temporal features were coded in Matlab. Segmentation of gestures based on skeletal joint data was implemented on Matlab and the classification based on the Random Forest was implemented using C++/Matlab. The SVM classifier was carried out using open-source Support Vector Machine libraries. Due to this fragmented nature of the implementation, the evaluation was conducted as a sequence of batch process.

5.3 Gesture Detection and Segmentation

We train a SVM model to classify a fixed length time window of each input sample and then use a sliding window on the test sample to obtain a probability distribution over time for each window. The predict labels for the test sample with likelihood scores average the prediction confidence of all the SVM models trained on the training samples. According with the predict labels of the test sample, we segment the sample into several candidate gestures. To tackle the problem of consecutive gestures in the input sample, we first manually annotate the frames of the training samples into two classes: *consecutive frame* and *nonconsecutive frame*. We then train a SVM model on these annotated frames and classify the frames of candidate gestures to *consecutive frame* and *nonconsecutive frame* to get the final segment of the samples.

5.3.1 Segmentation based on skeletal joints

We analyze the skeletal joint data stream from the Kinect sensor to identify the start frame and end frame of each gesture within an input sample. We approach this problem as two-class classification task: classify each frame of the input sample as *gesture* frame or *non-gesture* frame. The skeletal joint data provide world position, pixel position, and world rotation of 20 joints for each video frame. Since all the 21 gesture classes (20 pre-defined gesture plus 1 unrecognized gesture) are performed using the joints of upper body, we focus only on the joints above waist level reducing the number of joints from 20 to 12.

5.3.1.1 Skeletal Feature Engineering

We extract the skeletal feature from the skeletal joint data. The 3D coordinates of these joints are, however, not invariant to the position of the subject. Therefore we aligned the skeletal joints of each frame for each sample so that the hip centers of all frames are overlapped. We employ 3D position differences of joints to characterize gesture information including motion feature f_c and hand-based feature f_h. We extract features $f_{c,t}$ and $f_{h,t}$ from a 13-frame-long slide window s_t where the frame t is at the center of this slide window.

Let $p_{j,t} \in \mathbb{R}^3$ be the 3D world position $(x_{j,t}, y_{j,t}, z_{j,t})$ and of joint j at frame t. J represent the 12 joints used in our approach. The motion features $f_{c,t}$ of frame t are defined as the joints differences within the slide window s_t:

$$f_{c,t} = \{max(p_{j,i} - p_{j,t}) \mid \forall j \in J\}, i \in [t-6, t+6]; i \neq t \tag{5.1}$$

We designed the hand-based feature f_h to pay attention to hand motion signals as all the gestures are performed by the hands. In particular, we consider only the y-coordinate of the hand joint locations and hip joint locations. We first compute the y-coordinate differences between hand joint and hip joint:

$$\delta_{hh,i} = max(\mid y_{jr,i} - y_{jh,i} \mid, \mid y_{jl,i} - y_{jh,i} \mid) \tag{5.2}$$

where jr, jl, jh represent the right hand joint, left hand joints, and hip joint, respectively. As the same gesture can be performed by either right hand or left hand, Equation 5.4 is able to achieve the invariance under different hand performances. To capture the motion property of the hand joints, The hand-based features $f_{h,t}$ of frame t are defined as y-coordinate differences between hand joint and hip joint of each frame within the slide window s_t:

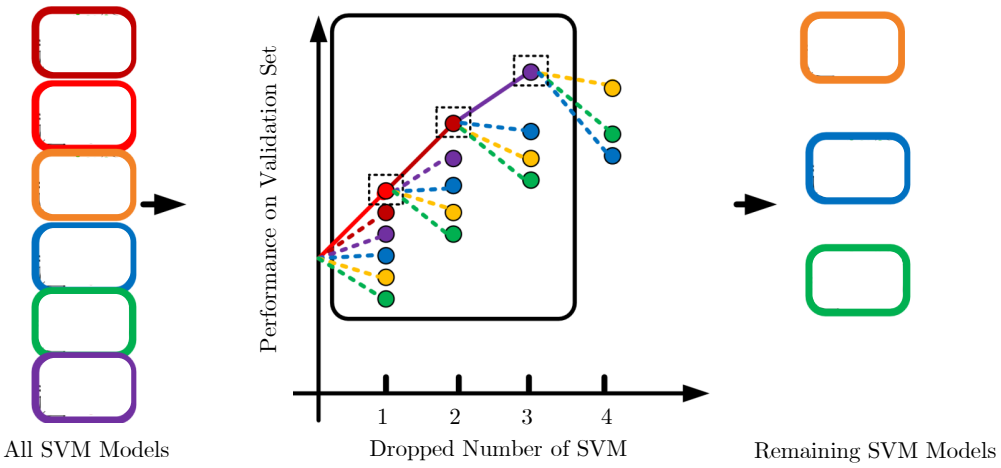


Figure 5.3: Illustration of the process of greedy SVM model selection described in Section 5.3.1.3: Left: initial number T of the SVM model ($T = 6$ in this figure). Middle: greedy SVM model selection process (the number of dropped SVM model is $n = 3$). Right: the remaining $T - n$ SVM model that maximize validation performance ($T - n = 3$)

$$f_{h,t} = \{\delta_{hh,i} \mid i \in [t-6, t+6]\} \tag{5.3}$$

5.3.1.2 Skeletal Feature Classifier

We extract the motion feature f_c and hand-based feature f_h from each frame of the input sample. In our implementation, each frame is represented by 13-frame-long slide window where the frame is at the center of the window. We annotate each frame with a label, either *gesture* frame or *non-gesture* frame according to the annotation labels provided by the training dataset. However, as the unrecognized gestures in the training dataset were mislabled as *non-gesture* frames, we choose the y-coordinate differences $\mid y_{jr,i} - y_{jl,i} \mid$ between right hand joint jr and left hand joint jl to filter out the flase *non-gesture* frames. Any *non-gesture* frame which has the y-coordinate differences $\mid y_{jr,i} - y_{jl,i} \mid$ above a specified threshold are removed from the training data. To eliminate the effect of different sizes of the performers, we train a two-class SVM model for each input sample of the training dataset and validation dataset, in total, having 500 SVM models.

5.3.1.3 Greedy SVM Model Selection

We use the samples to train 500 SVM models independently, however, some smaples include mislabled frames. Therefore, we need to select the best subset of SVM models based on the validation performance in a greedy manner. We select the subset of models in a sequential manner: first, we evaluate the performance of T SVM models on validation data. Then, we drop one SVM model and evaluate the performance on the remaining set of $T-1$. We select the $T-1$ SVM models that maximizes the validation performance. We then evaluate the validation performance when we drop one more from the $T-1$ SVM models and pick the $T-2$ SVM models that maximize performance. We repeat this process for n times, and select the best subset as the remaining $T-n$ SVM models that maximize the validation performance ($n = 3$ in Fig. 5.3). A greedy method is required as there are too many possible subsets of SVM models to enumerate exhaustively. Once the best subset of $T-n$ SVM models are selected, we test each sample in the testing dataset. If the number of the consecutive frames which are labeled as *gesture frames* is greater than a specified threshold, it is assumed that a candidate gesture is detected. This is done to filter away any impulse or spurious signals that might be mistaken as a candidate gesture.

5.3.2 Dealing with consecutive gestures

Normally, each sample include $10 \sim 20$ candidate gestures. Most of them are typically separated by *long-pauses* in between (e.g., the *long-pause* contains tens of *nongesture* frames), but some of them are consecutive gestures (e.g., separated by *short-pause* containing less than 2 frames) (see Fig. 5.4). The above SVM models may classify the *non-gesture* frames of *short-pause* as *gesture* frames. To tackle this problem, we train a new SVM model to classify the frames of candidate gestures as *consecutive frame* or *nonconsecutive frame*. To get the training data of the new model, we scan all the samples in the training and validatation dataset and find the consecutive gestures where two adjacent gestures are separated by a *short-pause*. We manually annotate the frames in the *short-pause* as *nonconsecutive frame* and the frames in the adjacent gestures as *consecutive frame*. We then train the SVM model based on the labeled training data. For the frames in the candidate gesture, if two consecutive frames are labeled as *consecutive frames* by the new SVM model, we divided the candidate gesture into another two candidate gestures further (indicated by the black circles in Fig. 5.4). Fig. 5.4 shows the segmentation results of Sample 701 and 707 in the testing dataset.

The advantage of the proposed detection and segmentation method based on skeletal joint data is that it efficiently identifies segments containing a candidate gesture without any heavy computation. The disadvantage is that it does not differentiate between pre-defined and unrecognized gestures. Consequently, segments containing unrecognized gestures get included in the segmentation results.

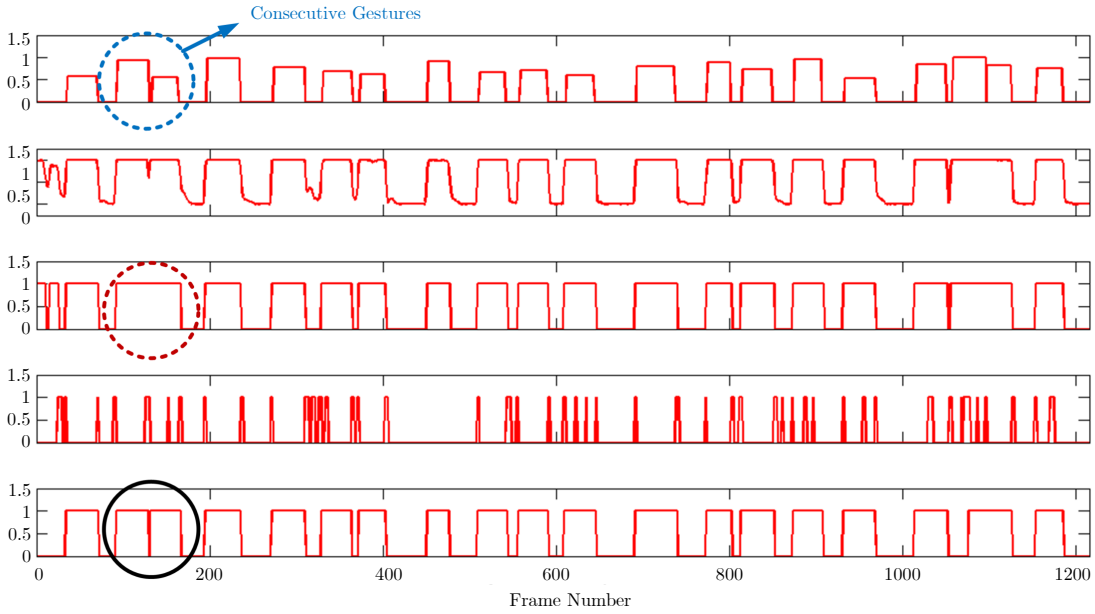
5.4 Gesture Classification

The segmentation results can not separate the pre-defined gestures from the unrecognized gestures. Thus, during the gesture classification phase, we will perform the classification of 21 classes of gestures (20 pre-defined gestures plus unrecognized gesture) instead of 20 classes of pre-defined gestures. We first explore a 3D dense representation of each candidate gesture. Dense feature have been shown the advantage in classifying human activities [164]. We use a random forest framework to identify discriminative video blocks. Inspired from [164], we combine discriminative training and randomization to obtain an effective classifier with good generalizability. For each tree node, we train an SVM classifier from one of the randomly sampled video blocks. This allows us explore a richer feature set efficiently as well as identifies semantically meaningful video blocks that closely match human intuition.

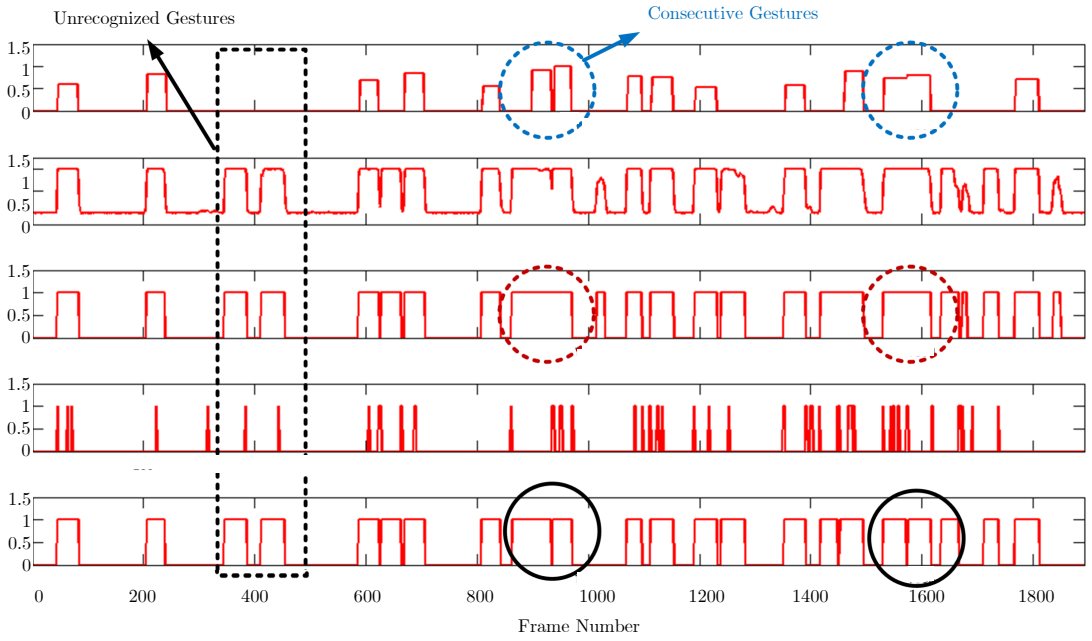
5.4.1 Spatio-temporal Feature Extraction

We extract spatio-temporal feature from four channels (grayscale, gradient, depth, surface normal) of RGB-D video data by using Independent Subspace Analysis (ISA) algorithm [85], which is based on the unsupervised learning framework in Chapter 3. ISA is a popular unsupervised learning algorithm that learns spatia-temporal features from unlabeled video data

The main advantage of unsupervised feature learning is that it readily applies to novel data, such as grayscale and gradient magnitude video data from a RGBD-camera. We learn spatio-temporal features up to four channels of RGB-D video data: grayscale, gradient, depth, and surface normal (z-axis). The learned features are visualized in Fig. 3.5. These features are interesting to look at and share some similarities. For example, the learned feature (each row of the sub-figure) is able to assign similar patterns into a group, and has sharper edges like Gabor filters.



(a) The detection result of sample 701



(b) The detection resutt of sample 707

Figure 5.4: Segmentation result of Sample 701 and Sample 707 in testing dataset. The first sub-figure shows the ground truth label of each sample. Peaks indicate a gesture being performed. The height of the peaks means different types of gestures. The second sub-figure shows the labeled results of the SVM models (without considering the consecutive gestures). The third sub-figure shows the initial segmentation results that filter away any impulse or spurious signals. The fourth sub-figure shows the labeled results of the SVM model for dealing with the consecutive gestures. The fifth sub-figure shows the final segmentation result of each sample.

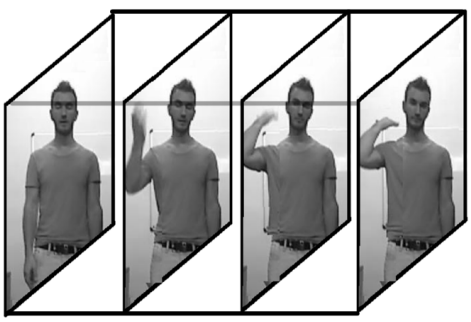
5.4.2 Spatio-temporal Dense Sampling Space

Our approach aims to identify discriminative spatio-temporal blocks that are useful for the gesture classification. For example, in order to recognize whether a human is performing a gesture *Ok*, we want to use the spatio-temporal blocks surrounding the human hands that are closely related to the gesture *Ok*. We need to identify not only the spatial position of this kind of blocks (the image coordinate of the blocks) but the temporal position of the blocks (the start and end timestamps of the blocks). An algorithm that can reliably locate such regions is expected to achieve high classification accuracy. We achieve this goal by searching over spatio-temporal blocks with arbitrary spatial size, temporal size, and the 3D position of the blocks. We refer to this extensive set of spatio-temporal blocks as the *spatio-temporal dense sampling space*, as shown in Figure 5.5. We densely sample spatio-temporal blocks with varying spatial and temporal size, and spatial and temporal position. The 3 varying dimensions of the sampling blocks include one dimension along with the width of the block, one dimension along with the height of the block, and one dimension along with the temporal length of the block. This figure has been simplified for visual clarity that all the dense sampled blocks starts from the first frame. The actual density of spatio-tempral blocks is significantly higher than the illustration in Figure 5.5.

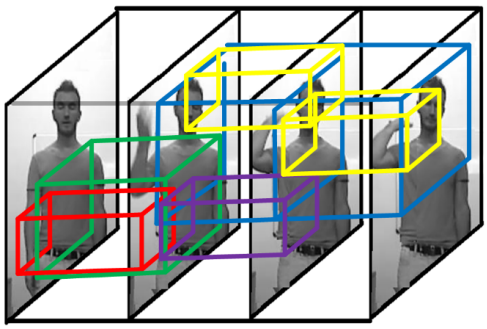
Sampling a fixed size $30pixels \times 30pixels \times 8frames$ of spatio-temporal blocks in a $300pixels \times 300pixels \times 50frames$ video every two pixels and two frames leads to thousands of spatio-temporal blocks. This increases many folds when considering blocks with arbitrary spatial and temporal sizes. Furthermore, there is much redundancy in this huge feature set. Richer feature indeed provide enough information for the classification task, however, many spatio-temporal blocks are not discriminative for distinguishing different gesture classes. Additionally, dense sampling introduces many overlapped spatio-temporal blocks which brings significant redundancy. Therefore, it is challenging to explore this 3D dimensional, noisy and redundant feature space. In this work, we address this problem using the idea of combining discrimination and randomization.

5.4.3 Discriminative Random Forest Framework

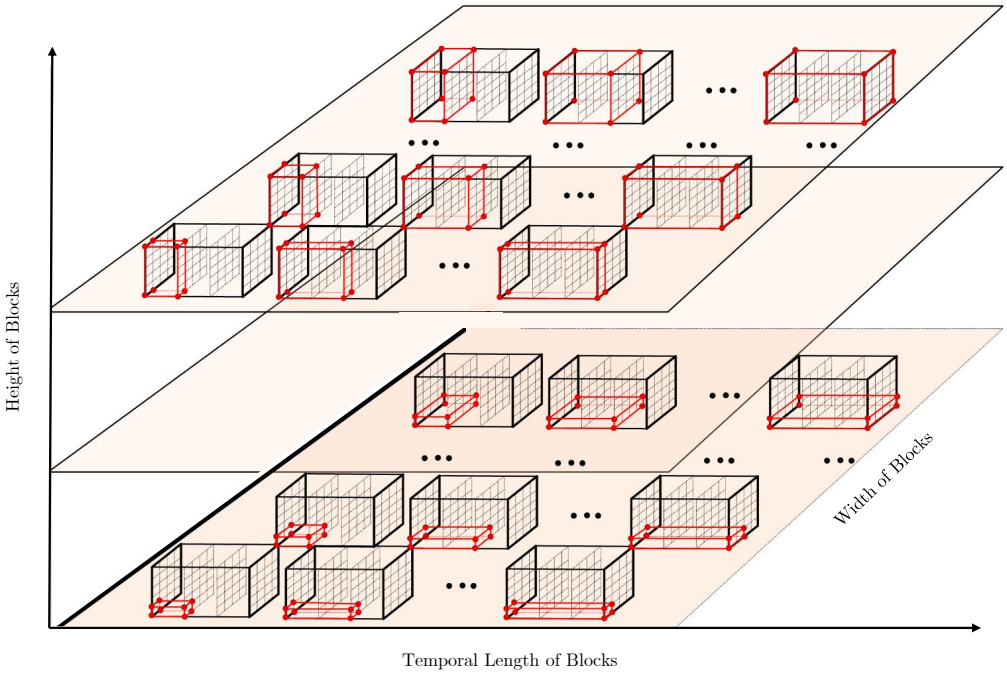
In order to explore the 3D dense sampling feature space for the gesture classification, we combine two ideas: 1) Discriminative training to extract the information in the spatio-temporal blocks effectively; 2) Randomization to explore the 3D dense feature space efficiently. Specifically, we



(a) Video sample



(b) Sampling spatio-temporal blocks



(c) The varying dimensions of the sampling blocks

Figure 5.5: Illustration of the proposed 3D spatio-temporal dense sampling space. (a) A sample of the video. (b) We densely sample spatio-temporal blocks with varying spatial and temporal size, and spatial and temporal position. (c) The 3 varying dimensions of the sampling blocks include one dimension along with the width of the block, one dimension along with the height of the block, and one dimension along with the temporal length of the block. This figure has been simplified for visual clarity that all the dense sampled blocks starts from the first frame. The sampled blocks are highlighted by red cuboids.

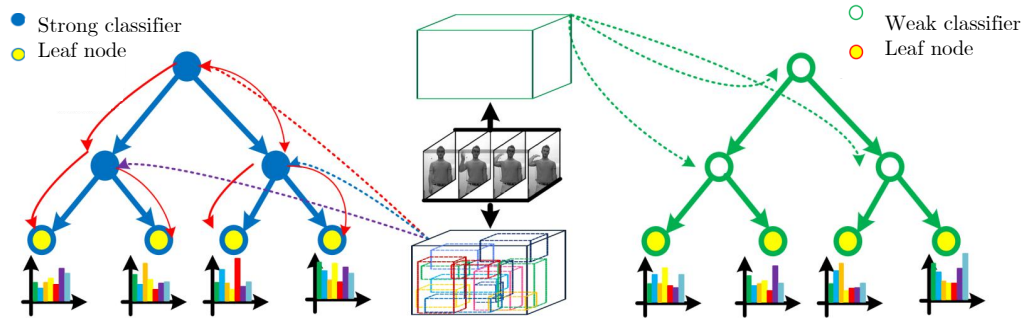


Figure 5.6: Comparison of our discriminative decision tree (Left side of the figure) with conventional random decision tree (Right side of the figure). Conventional decision trees use information from the entire video data at each node, which encodes no spatio-temporal information, which our decision trees sample the spatio-temporal blocks from the dense sampling spcace. The histograms below the leaf nodes illustrate the posterior probability distribution. Our approach use strong classifiers (SVM) in each node, while the conventional method uses weak classifiers.

adopt a random forest framework [164, 165] where each tree node is a SVM classifier that is trained on one spatio-temporal block.

5.4.3.1 Introduction of Random Forest Framework

A random forest is a multi-class classifier consisting of an ensemble of decision trees where each tree is constructed via some randomization. As illustrated in Fig. 5.6, the leaf nodes of each tree encode a distribution over the gesture classes. All internal nodes contain a binary classifier that splits the data into two parts and sends the two parts to its children nodes. The splitting is stopped when a leaf node is encountered. A candidate gesture is classified by descending each tree and combining the leaf distributions from all the trees of the forest. This method allows the flexibility to explore a rich feature space effectively because it only considers a small subset of features (e.g., several hundreds of spatio-temporal blocks sampled from the video data) in every tree node.

Each tree returns the posterior probability of an test example belonging to the given classes. The posterior probability of a particular class at each leaf node is learned as the proportion of the training videos (each training video contains one gesture) belonging to that class at the given leaf node. The posterior probability of class c_m at leaf l of tree t is denoted as $P^m_{t,l}(c_m)$, where m means the type of the modality used in the representation of video data. Thus, a test candidate gesture can be classified by averaging the posterior probability from all the trees of

the forest:

$$\hat{c}_m = \arg\min_{c_m} \frac{1}{T} \sum_{t=1}^{T} P_{t,l_f}^{m}(c_m) \tag{5.4}$$

where $\hat{c}_m$ is the predicted labeled using the modality data m, T is the number of the trees of the forest, and l_f is the leaf node that the testing video falls into. To fuse multi-modality data in the random forest framework, we utilize later fusion to classify the test candidate gesture:

$$\hat{c} = \arg\min_{c} \frac{1}{M \times T} \sum_{t=1}^{T} \sum_{m=1}^{M} P_{t,l_f}^{m}(c_m) \tag{5.5}$$

where $\hat{c}$ is the predicted labeled using the multi-modality data, M is the number of the types of the modality data.

5.4.3.2 Sampling the Spatio-temporal Dense Feature

As shown in Fig. 5.6, each internal node in the decision tree corresponds to a single spatio-temporal video blocks that are sampled from the 3D dense sampling space (Section 5.4.2), where the spatio-temporal blocks can have many possible spatio-temporal size and spatio-temporal positions. In order to sample a candidate spatio-temporal blocks, we first normalize all videos to unit width, height and temporal dimension, and then randomly sample (x_l, y_l), (x_r, y_r) and (t_s, t_e) from a uniform distribution $U([0, 1])$. The coordinates (x_l, y_l) and (x_r, y_r) specify two diagonally opposite vertices of the spatial region of the block. The coordinates (t_s, t_e) specify the start and end position along the temporal dimension of the block. Such blocks could correspond to small area of the gesture segment or even the complete gesture segment. This allows the method to capture both the global and local information in the video.

In our approach, each sample saptia-temporal block is represented by a histogram of spatio-temporal features. The features are augmented with the decision value $w^T f$ (described in Equation 5.6) of this video segment from its parent node (indicated by the red lines in Fig. 5.6). Therefore, the feature representation combines the information of all upstream tree nodes that the corresponding video segment has descended from.

5.4.3.3 Learning the binary classifier of the tree node

We describe the process of learning the binary splits of the data using SVM. This is achieved in two steps: 1) Randomly assigning all segments from each class to a binary label; 2) Using SVM to learn a binary split of the data. Assume that we have C classes of gesture segments at

a given node. We uniformly sample C binary variables, b. We then assign all sampled blocks of a particular class c_i a class label of b_i. As each node performs a binary split of the data, this allows us to learn s simple binary SVM at each node. Using the feature representation f of an spatio-temporal block, we find a binary split of the data:

$$score = w^T f, \begin{cases} score \leq 0, \ go\ to\ left\ child \\ score > 0, \ go\ the\ right\ child \end{cases} \tag{5.6}$$

where w is the set of weights learned from a linear SVM.

We evaluate each binary split that corresponds to an spatio-temporal blocks with the information gain criteria [163], which is computed from the complete training video segments that fall at the current tree node. The splits that maximize the information gain are selected and the splitting processing is repeated with the new splits of the data. The tree splitting stops if a pre-defined tree depth or a minimum number of samples in the current node has been reached, or the information gain of the current node is larger than a specified threshold.

5.4.4 Pre-processing and Implementation Details

5.4.4.1 Pre-processing of the RGB-D video data

It is our observation that gestures only related to upper body movement of the performers. Within the performance of the gestures of each sample, there is little movement of the lower part of the body, especially the foot movement. Therefore, we cut out part of the video data containing only the upper body of the performers from the entire video data. During the gesture classification phase, we extract spatio-temporal features from this partial video instead of the complete video in each sample. We resize this partial video to a fixed spatial size video of 200×200. For the learning of the binary split of the tree node, the randomly sampled spatio-temporal blocks of different gesture segments should have the same spatio-temporal size and spatio-temporal positions. However, the temporal dimension of gesture segments is different. We therefore employed time normalization for the temporal alignment of all gesture segments. We apply the max pooling along the temporal dimension of the dense sampling feature space of the gesture segments.

5.4.4.2 Implementation details

We densely extract four types of ISA features (gray-ISA, depth-ISA, gradient-ISA, and normal-ISA feature) on each gesture segments with a spatial spacing of 2 pixels and a temporal spacing

of 2 frames. Using k-means clustering, we construct a vocabulary of codewords for each modality. We use Locality-constrained Linear Coding [166] to assign the spatio-temporal features to codewords instead of vector quantization coding. In order to achieve good performance on classification tasks, traditional spatial-pyramid matching approach requires nonlinear classifiers. For locality-contrained linear coding, each descriptor is projected into its local coordinate system by utilizing the locality constraints, and the final representation is generated by integrating the projected coordinates by max pooling. With a linear classifier, locality-constrained linear coding performs remarkably better than nonlinear spatial pyramid matching, achieving state-of-the-art performance on image classification tasks [166]. In this work, we use a linear SVM as stated in Section 5.4.3.3. A bag-of-words histogram representation of the spatio-temporal blocks is used if the spatial size and temporal size of the blocks are smaller than 0.2, while a 2-level spatial pyramid is used if the spatial size of the block is between 0.2 and 0.9. We limit the maximum spatial size and temporal size to 0.9 and 0.8 respectively. For each tree of the forest, we sample 150 spatio-temporal blocks in the root node and the first level nodes respectively, and sample 200 spatio-temporal blocks in all other nodes. Sampling a smaller number of blocks in the root can reduce the correlation between the resulting trees. In total, we have trained 100 trees for each type of ISA features.

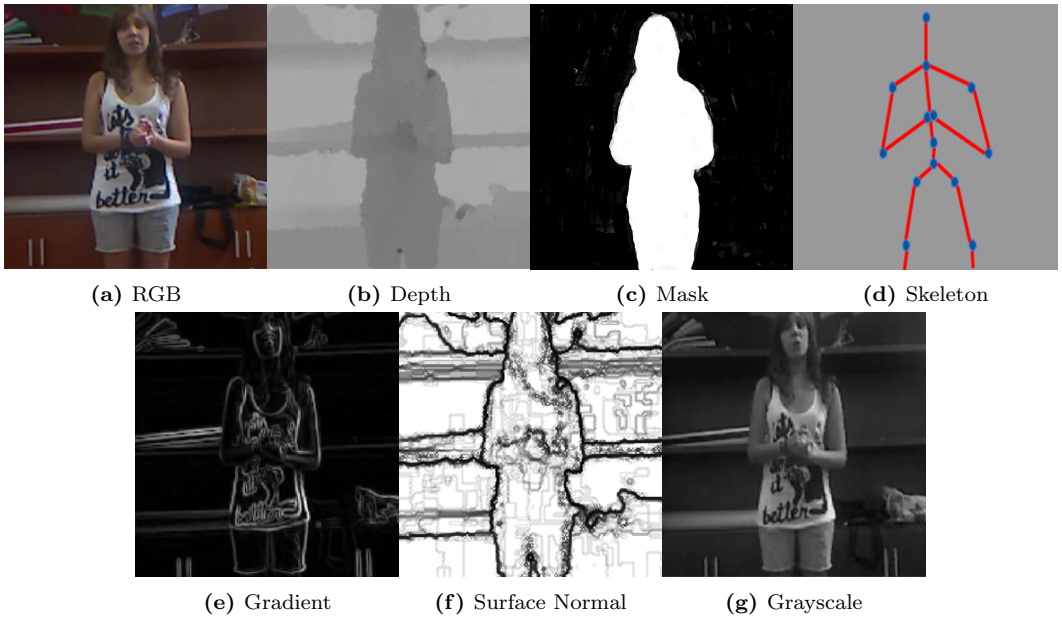
5.5 Results and Discussion

In this section, we present the experimental results to evaluate the performance of our approach. We use the training set and validation set as the final training dataset, and the testing set as the final testing dataset in the following experiments. To best understand the classification performance of our approach, we use the ground truth labels to segment the testing dataset instead of the predicted gesture segmentations.

5.5.1 Chalearn Gesture Dataset

Chalearn Gesture Dataset is used to evaluate our approach, which is originally created for Chalearn gesture spotting challenge. The aim of this challenge on 'Multiple instance, user independent spotting' of human gestures. This means that gestures should be recognized from multiple instances for each class performed by different users.

The dataset is captured by Kinect, including RGB video data, depth video data, segmentation mask video data, skeletal joint data. In addition to these four modalities, we also create



(a) RGB **(b)** Depth **(c)** Mask **(d)** Skeleton

(e) Gradient **(f)** Surface Normal **(g)** Grayscale

Figure 5.7: The Chalearn Gesture dataset is captured by Kinect, including RGB video data, Depth video data, video data of user segmentation mask, skeletal joint data. In addition to these four modalities, we also create two new types of modalities data, which are surface normal video and gradient video data(best viewed in color).

(a) Vattene (b) Viene qui (c) Perfetto (d) E un furbo (e) Che due palle

(f) Che vuoi (g) Vanno d accordo (h) Sei pazzo (i) Cos hai combinato (j) Nonme me friega niente

(k) Ok (l) Cosa ti farei (m) Basta (n) Le vuio prendere (o) Non ce ne piu

(p) Ho fame (q) Tanto tempo (r) Buonissimo (s) Si sono messi (t) Sono stufo

Figure 5.8: 20 Gesture examples in Chalearn Gesture Dataset.

three new types of modalities data, which are surface normal video data, gradient video data and grayscale video data(See Fig. 5.7). The Chalearn gesture dataset includes 940 long video sequences. Each of the video are performed by a single person. More than 14,000 gestures are drawn from a vocabulary of 20 Italian sign gesture categories from these video sequences, which are *vattene, vieniqui, perfetto, furbo, cheduepalle, chevuoi, daccordo, seipazzo, combinato, freganiente, ok, cosatifarei, basta, prendere, noncenepiu, fame, tantotempo, buonissimo, messidaccordo, sonostufo*. Example of the gestures in the Chalearn gesture dataset are show in Fig. 5.8 The number of samples are well balanced between classes. The average length of gestures is 39 frames, the maximum frame number is 104 and the minimum frame number is 16. The input samples include other unrecognized gestures that are not included in the vocabulary. This dataset is challenging due to the 'user independent' setting. Different gesture classes differ slightly in hand pose while the arm movements of these gestures are almost the same. The execution of the same gesture class by different users differs in the spend, location and range.

5.5.2 Evaluation

We trained our models on 10000 gesture segments (20 pre-defined gesture classes + one unrecognized gesture classes) of the training dataset. Because of the utilization of ground truth labels, we performed the classification task of 20 pre-defined gesture classes on 3579 gesture segments in the testing dataset. We used four channels (grayscale, depth, gradient magnitude, surface normal) of the RGB-D video data to train the spatio-temporal features and the discriminative random forest models. Finally, we get four types of spatio-temporal features (Gray-ISA, Depth-ISA, Gradient-ISA, Normal-ISA), and four RF models (Gray-ISA-Drf, Depth-ISA-Drf, Gradient-ISA-Drf, Normal-ISA-Drf) where each model contains 100 decision trees. We also utilize a fusion model which use a simple late fusion strategy by combining the likelihood scores of the above four RF models.

The classification results measured by mean average precision (*map*) and average accuracy (*acc*) are shown in Table 5.1. The Gray-ISA-Drf model achieves the best result on the average *map* (95.3%) and *acc* (90.3%) of 20 gesture classes. Note that we achieved this accuracy using very-low resolution videos (200 $pixels \times 200$ $pixels$). In detail, the Gray-ISA-Drf model and fusion model achieve the best result on seven and ten out of 20 classes, respectively. While the performance based on the Gray-ISA-Drf/Depth-ISA-Drf/Gradient-ISA-Drf models is promising, the accuracy of the Normal-ISA-Drf model is relative low. This is probably because the process of down-sampling depth video to a lower resolution loses some important information of surface

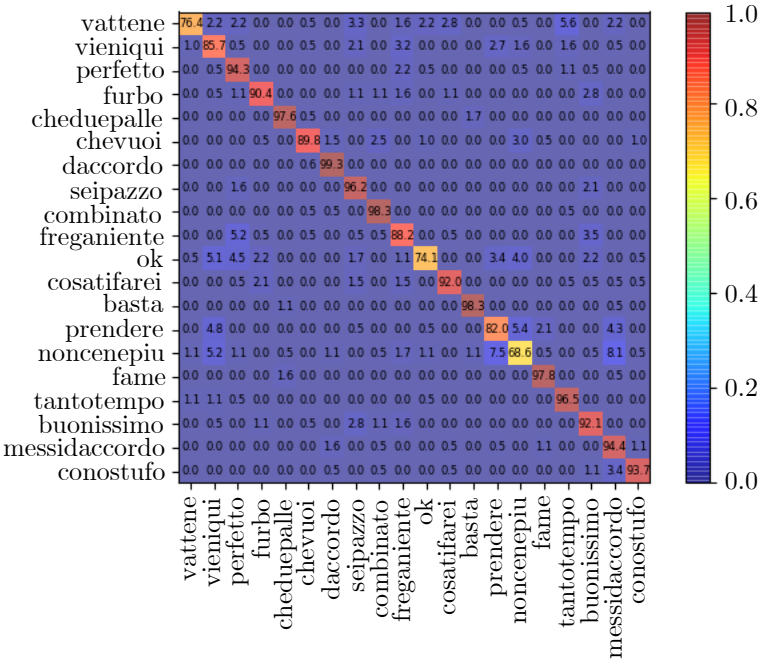
Table 5.1: Mean average precision (map) and classification accuracy (acc) on the testing data of *ChaLearn Looking at People* Challenge (Track: 3). The Gray-ISA-Drf, Depth-ISA-Drf, Gradient-ISA-Drf, Normal-ISA-Drf and Fusion model were represented by Gray, Depth, Gradient, Normal, Fusion in this table, respectively. Each column shows the results obtained from one model. The best result is highlighted with bold fonts.

Gesture	Gray		Depth		Gradient		Normal		Fusion	
	map	acc	map	acc	map	acc	map	acc	map	acc
vattene	**90.1**	76.4	88.2	73.6	87.2	78.1	81.8	68.5	88.5	**78.6**
vieniqui	**92.1**	**85.7**	87.6	79.1	86.6	75.3	79.0	68.1	90.0	81.3
perfetto	**94.7**	**94.4**	92.7	89.3	92.8	90.4	90.1	87.6	93.1	90.4
furbo	**97.8**	90.4	91.5	92.7	95.9	92.1	90.3	91.6	96.4	**93.8**
cheduepalle	**99.6**	97.7	**99.6**	97.7	99.1	96.5	99.2	**98.8**	**99.6**	98.3
chevuoi	96.1	**89.9**	**96.4**	85.9	93.5	83.8	94.3	81.3	**96.4**	86.9
daccordo	99.3	99.4	99.2	97.5	98.8	98.8	97.7	93.9	**99.6**	**100**
seipazzo	97.5	**96.2**	96.8	92.4	96.3	90.3	94.8	90.8	**97.6**	94.6
combinato	**99.0**	98.3	97.3	97.3	98.5	97.3	98.1	97.3	98.8	**98.4**
freganiente	**92.7**	**88.2**	87.4	75.9	88.2	82.9	81.1	67.0	90.8	82.4
ok	**88.4**	**74.1**	81.5	60.9	84.8	65.5	79.6	40.2	88.0	66.7
cosatifarei	**96.4**	92.0	95.9	90.4	94.8	91.5	92.5	89.4	**96.4**	**92.6**
basta	99.8	98.3	99.8	**99.4**	99.7	98.3	99.6	98.8	**99.8**	98.9
prendere	**93.1**	82.1	89.5	81.5	91.4	**83.7**	84.2	72.3	91.9	80.4
noncenepiu	**83.8**	68.6	75.2	70.9	76.0	60.5	62.9	60.5	80.2	**71.5**
fame	99.0	97.8	99.0	97.8	98.7	94.6	98.3	97.8	**99.1**	**98.4**
tantotempo	**99.0**	96.5	96.9	95.4	98.5	96.5	96.1	96.5	98.4	**97.7**
buonissimo	**94.0**	**92.1**	85.7	77.5	88.7	82.6	83.5	75.8	90.5	84.8
messidaccordo	96.8	94.4	97.4	95.6	94.1	96.7	95.4	91.7	**97.8**	**97.8**
sonostufo	98.0	93.7	98.8	93.7	96.8	85.7	95.6	89.1	**98.8**	**94.3**
	95.3	**90.3**	92.8	87.2	93.0	87.1	89.7	82.9	94.6	89.4

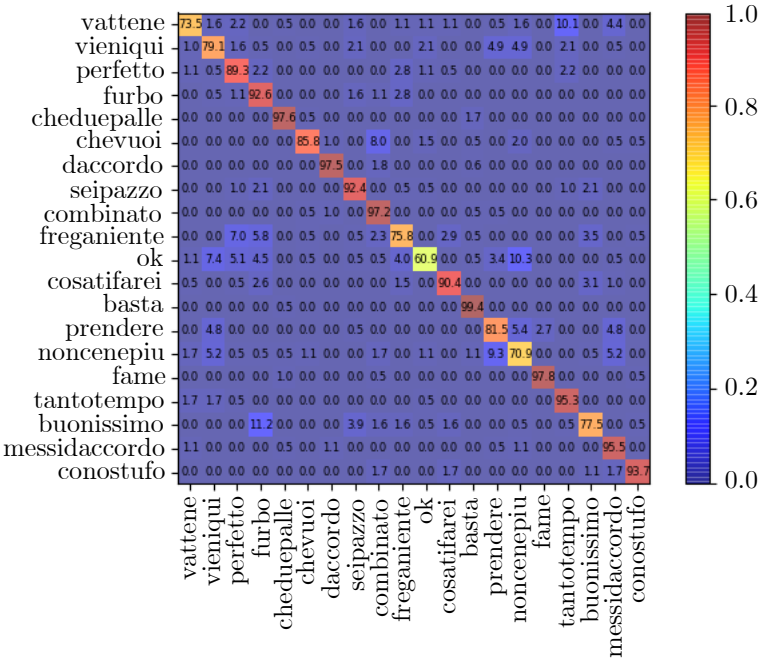
normals. In addition, the fusion model decreased the performance compared with the Gray-ISA-Drf model. It is expected to achieve a better performance by investigating different fusion strategies (e.g., different combination of single models, fusion before training the random forest model). Fig. 5.9, 5.10 are the visualization of confusion matrix of the Gray-ISA-Drf model. We can see that 12 out of 20 gesture classes achieved a result of $> 90\%$ accuracy. This is a good performance considering that we use single spatio-temporal feature, without using any hand-engineering spatio-temporal features or skeleton-based feature (for classification task).

In Fig. 5.11, we visualize the 2D heat map of the dominant positions of the first 40 gesture segments in the testing dataset. The 2D heat map shows the distribution of the discriminative positions discovered by our approach for the specific gesture segment. The 2D heat maps are obtained by aggregating the spatial region of the spatio-temporal block of all the tree nodes in the random forest weighted by the probability of the corresponding gesture class. We can see the difference of distributions for different gesture classes. We observe that they show semantically meaningful locations of where we would expect the discriminative regions of subjects performing different gestures to occur. For example, the regions of corresponding to the hand joint of the human body are usually highlighted. We can also see that the regions corresponding to background or irrelevant joints (e.g., head, hip center) are not frequently selected.

In Fig. 5.12, we visualize the 3D heat map of the dominant spatio-temporal positions of the first 9 gesture segments in the testing dataset. The 3D heat maps are obtained by aggregating the spatio-temporal space of the spatio-temporal blocks of all the tree nodes in the random forest weighted by the probability of the corresponding gesture class. To a better visualization, we mapped the 3D heat maps to a sequence of 2D heat maps where the timestamps of the heat maps range from the start of the gesture segment to the end of the gesture segment. From this figure, we can clearly see that the different timestamps of the gesture segment have completely different heatmaps of the dominant positions. This means that, at different phase of a gesture, we would expect the different discriminative regions of the subjects performing gesture to occur. In addition, the 3D heat maps show three distinct phases of a gesture: pre-stroke, nucleus, post-stroke (see Fig. 5.12), which consists with description in previous research on hand gesture [167]. The pre-stroke corresponds to the subject moving from the resting posture to the initial posture, which matches the start-phase of our 3D heat maps. During this phase, the spatio-temporal spaces are not frequently selected by our model (indicated by the blue space in the start-phase of the 3D heat map). The nucleus corresponds to the actual gesture performed by the subject, which matches the middle phase of our 3D heat maps (indicated by the red space in the middle

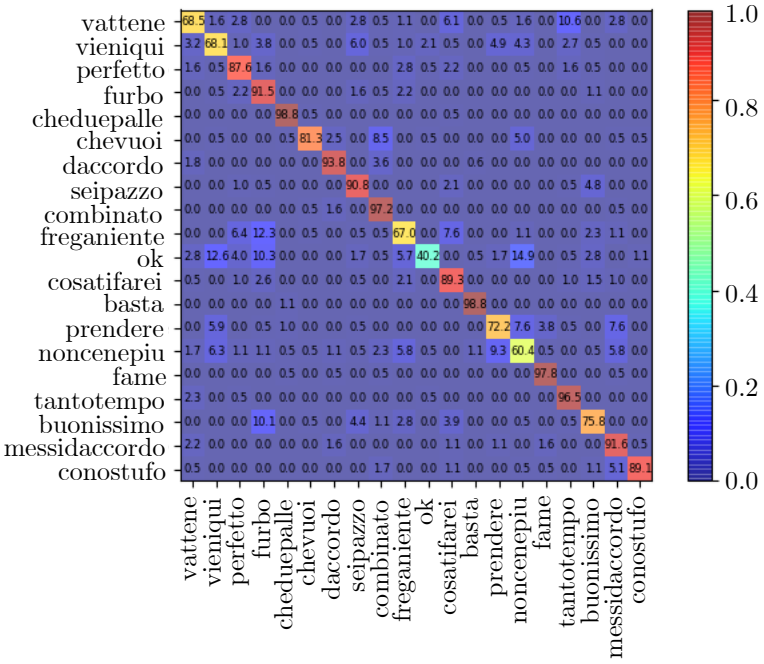


(a) Gray-ISA-Drf Model

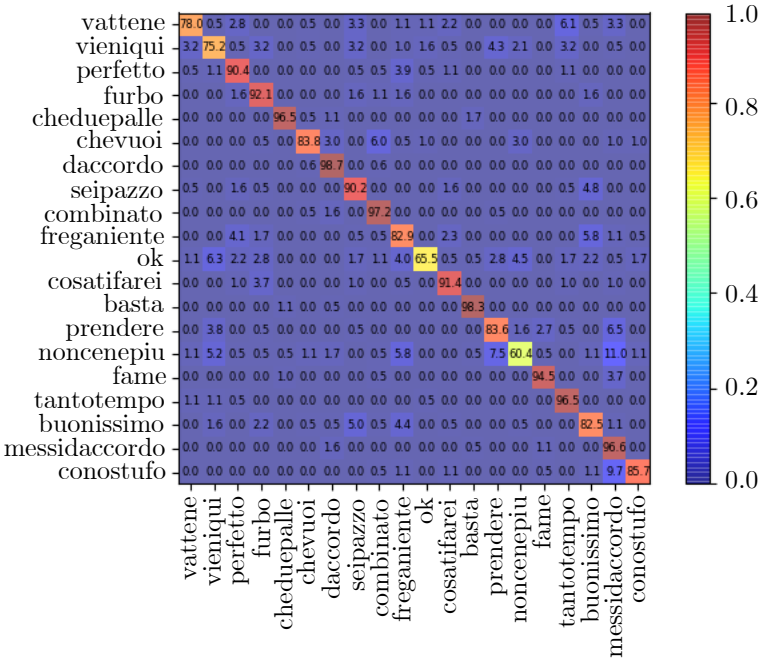


(b) Depth-ISA-Drf Model

Figure 5.9: The fusion matrice on the testing dataset using the Gray-ISA-Drf model and Depth-ISA-Drf model. Rows represent the actual gesture classes, and columns represent predicted classes (best viewed in color).



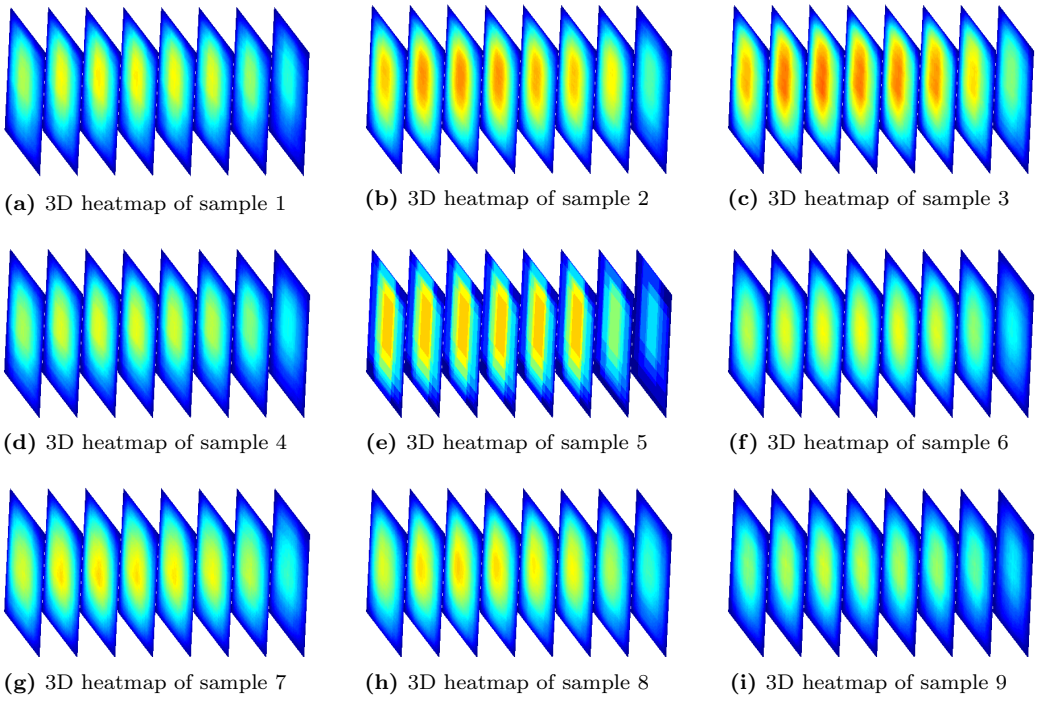
(a) Gradient-ISA-Drf Model



(b) Normal-ISA-Drf Model

Figure 5.10: The fusion matrice on the testing dataset using the the Gradient-ISA-Drf model and Normal-ISA-Drf model. Rows represent the actual gesture classes, and columns represent predicted classes (best viewed in color).

Figure 5.11: The 2D heat maps of the dominant positions of the first 40 gesture segments in the testing dataset. Each 2D heat map is corresponding to one gesture segment. The 2D heat maps are obtained by aggregating the spatial region of the spatio-temporal block of all the tree nodes in the random forest weighted by the probability of the corresponding gesture class. Red rectangles mean the mis-classified gesture segments. Red indicates high frequency and blue indicates low frequence (best viewed in color).



(a) 3D heatmap of sample 1 **(b)** 3D heatmap of sample 2 **(c)** 3D heatmap of sample 3

(d) 3D heatmap of sample 4 **(e)** 3D heatmap of sample 5 **(f)** 3D heatmap of sample 6

(g) 3D heatmap of sample 7 **(h)** 3D heatmap of sample 8 **(i)** 3D heatmap of sample 9

Figure 5.12: The 3D heat map of the dominant spatio-temporal positions of the first 9 gesture segments in the testing dataset. The 3D heat maps are obtained by aggregating the spatio-temporal space of the spatio-temporal blocks of all the tree nodes in the random forest weighted by the probability of the corresponding gesture class. To a better visualization, we mapped the 3D heat maps to a sequence of 2D heat maps where the timestamps of the heat maps range from the start of the gesture segment to the end of the gesture segment. The left side of each 3D heat map is the start point of the gesture, and the right side of the 3D heat map is the end point of the gesture. Red indicates high frequence and blue indicates low frequence (best viewed in color).

part of the 3D heat map). Post-stroke corresponds to the hand motions from the end of the gesture back to the resting posture, which matches the end of the 3D heat maps (indicated by the blue space at the end of the 3D heat map).

5.6 Conclusion

Gesture detection and recognition has a wide variety of applications and the Chalearn gesture dataset serves as an important benchmark of the state-of-the-art in this field. We present our multi-modality gesture detection and recognition system in this chapter. Comparing to state-of-the-art approaches relying on hand designed features, our system is able to extract spatio-temporal features from four different modalities of RGB-D data in an unsupervised way.

5. MULTIMODAL GESTURE DETECTION AND RECOGNITION WITH RANDOMIZATION AND DISCRIMINATION

In addition, our system utilizes the random forest framework with discriminative decision trees to discover spatio-temporal blocks that are highly discriminative for gesture recognition tasks. We show that our method identifies semantically meaningful spatio-temporal blocks that closely match human intuition.

Chapter 6

Conclusion

In the concluding chapter of this thesis, we take a retrospective look to get an overview of the proposed work and to discuss the contributions and future work to the field of 3D human motion perception.

In Section 6.1, we summarize this thesis and conclude the main contributions. After that, in Section 6.2, we briefly revisit thesis goals which are previously introduced in Chapter 1. We discuss how we finally achieve these goals. Finally, in Section 6.3, we provide a vision of futurity for developing 3D human motion perception system.

6.1 Summary

In this thesis, we consider the task of developing 3D human motion perception system where human motions are recorded in multi-modal time series data. Human motion is one of the most natural means for humans to regulate interactions with the environment. Machine recognition of human motions constitutes an active research field due to its various applications in robotics, automotive industry, video surveillance and human-machine interaction. The interest in the topic is also motivated by the increasing popularity of novel 3D sensing devices. The low-cost 3D sensors enable researchers to record new 3D datasets and to design novel representation and algorithms. While 3D human motion perception may seem trivial (e.g., with the help of skeleton data), it is obviously incorrect. The key factor is that 3D muli-modal data and 2D modality data differ in essence. First, 3D modality provide 3D structure information, but there are some technical limitations such as low resolutions of the 3D data. Second, 3D modality data are full of noisy, and tracking results of the skeleton may be wrong if occlusions occur. Third, multi-modality data make the computation more expensive. Therefore, directly extending the

solutions of 2D human motion perception into 3D space is inefficient. In this thesis, we are developing efficient 3D human motion perception system by introducing advanced machine learning techniques.

In Chapter 2, we provided a comprehensive study of state-of- the-art 3D human motion perception methodologies and discuss the advantages and disadvantages of different methods. While a significant amount of research in the literature is described and analyzed in detail, we did not intend to cover all work in this area. We focused on recent work in machine recognition of human motions based on 3D video datasets. In this chapter, we observed that in 3D human motion perception there are some challenges which are not fully addressed by state-of-the art methodologies: (1) Multi modalities challenge. The depth cameras provide multi modalities, such as skeleton joint data, depth data, gray-scale data, surface normal data. To achieve a robust and efficient human motion perception system, how to represent multi modalities in an efficient way? (2) Domain-dependent and modality-dependent problem. One drawback of state-of-the-art human motion perception systems is that they are highly domain-dependent and modality-dependent. Previous study shows that there is no universally best hand-engineering feature for different dataset. (3) Redundant knowledge. In many computer vision tasks, dense feature representations are often used to capture enough information from high-dimensional visual data. This principle can also be employed in 3D human motion perception. However, rich representations always introduce significant redundancy among the feature space, and are not discriminative for distinguishing different human motions. Based on these observations, we firstly developed an unsupervised feature learning framework which we introduced in Chapter 3. We described our approach for learning 3D spatial-temperal feature from RGBD video data and 3.5D spatial-temperal feature from RGBD video and skeleton data. As a general framework, it can be used by any discriminative and generative classifiers in human motion perception systems.

After introducing the unsupervised learning feature framework, in Chapter 4, we present an efficient 3D action recognition system by combining unsupervised feature learning with discriminative feature mining. Unsupervised feature learning allowed us to extract spatio-temporal features from unlabeled video data. With this, we can avoid the cumbersome process of designing feature extraction by hand. We proposed an ensemble approach using a discriminative learning algorithm, where each base learner is a discriminative multi-kernel-learning classfier. Our approach was able to learn an optimal combination of joint-based features. In our evaluation, we analyzed the efficiency of our 3D action recognition approach. In more detailed

discussions, we investigated which joint subsets are discriminative for different types of actions, and we studied which of these joints are sufficient to recognize these actions. Our experimental results of the EnMkl approach showed a performance superior to existing techniques. Results also suggested that learning spatio-temporal features directly from depth video data may be a promising direction for future research, as combining these features with ensemble learning may further increase performance.

In Chapter 5, we introduced a novel human motion perception system called *multimodal gesture detection and recognition system with randomization and discrimination*. We mainly focused on human gesture recognition which is regarded as a sub-ordinate level categorization problem. Sub-ordinate level categorization problem did not receive too much attention in computer vision community. We tackle this problem by combining two key concepts: *randomization* and *discrimination*. A *discriminative random forest* for the gesture recognition was built in our system. Unlike traditional random forest, *discriminative random forest* was built by discriminative decision trees. Our approach was able to efficiently mine a very dense sampling 3D space by using strong classifiers at each node and combining information along the different depth of the tree. Comparing to the weak classifier in classical random forest, strong classifiers significantly enhanced the power of the decision trees. At the same time, the dense sampling of 3D space guarantee low correlation of different trees. The evaluation results showed the ability to discover meaningfully semantic information from a large number of video blocks with arbitrary sizes, shapes, and locations in space and time, as well as discover pairs of video blocks that carry discriminative video data statistic.

6.2 Discussion

The efficiency of 3D human motion perception system is the leading principle of this thesis. Efficiency is viewed from three sides in this thesis: the perception system gets more efficient if it processes data automatically without human intervention while giving the high performance results, the perception system gives better result within an optimized combination of multiple input modalities, and the perception system achieves meaningful results by discovering the useful knowledge from a richer representation.

Efficiency is realized by developing advanced machine learning techniques in this thesis. Advanced machine learning refers to the process of generating and mining knowledge from empirical example data, collected by observing a process or system of interest. In this thesis,

techniques are mainly explored to cope with the noisy and ambiguous data provided by the depth cameras, and to learn a more robust, effective and discriminative feature representation of the 3D human motions over richer collections. This thesis shows that the developed advanced machine learning techniques guarantee the efficiency of each key procedures of 3D human motion perception.

Efficiency is validated by combining three concepts in this thesis: un-supervision, randomization and discrimination. Un-supervision is seen as a way of learning and extracting features directly from the unlabeled video data. It leverages the plethora of the unlabeled data and adapt easily to new modality data. Randomization is seen as a way of discovering meaningful knowledge over the rich collections of features. It enables the algorithm to efficiently mine a very dense spatio-temporal sampling space of the video data in a principled way. Discrimination is seen as a way of learning the important and discriminative characteristics of feasible human motions from labeled video data. Together with un-supervision and randomization, it guarantees the effectiveness and efficiency of the algorithms.

Efficiency is demonstrated by two main topics: 3D human action recognition and 3D human gesture recognition in this thesis. The first topics is about the recognition of the 3D human actions by using unsupervised learning and ensemble learning techniques. Unsupervised feature learning allows us to extract spatio-temporal features from the depth video data. It avoids the cumbersome process of manually designed feature extraction. An ensemble approach using a discriminative learning algorithm is developed to learn an optimal combination of joint-based features. The evaluation includes a comparison to state-of-the-art methods on the MSRAction 3D dataset. In the second part a 3D human gesture detection and recognition system is built under the random forest framework by combining discriminative training and randomization. The system aims at the recognition of continuous, natural human gestures with the multi-modality nature of the visual cues, as well as technical limitations such as spatial and temporal resolution and unreliable depth cues. The evaluation is performed on the world's largest 3D human gesture dataset Chalearn Gesture dataset.

6.3 Future Work

An important future work is to fusion multi modality video representations. In this thesis, we discussed 3 categories of 3D video representations. All of them develop techniques on a single modality. The performance relies on the accuracy of the modality data. This issue should be

addressed by multi-modality fusion representations, which addresses the hierarchical structure of different types of features.

Another future work is to pay more attention to high-level vision tasks. Most of the classification models focus on low-level action tasks, where the video is already divided into segments which contain one class of a predefined action label. The action localization task is thus ignored. Such approaches have many limitations when applied to a realistic scenario where segementation in space and time is impossible. Due to the large scale and complex properties of the high-level vision tasks, most of the approaches are complicated and hard to generalize. It is expected that more research addresses this topic and develops algorithms and implementations to promote promising solutions in this topic.

A very important issue in vision based perception system in general is achieving real time performance. Many 3D action recognition algorithms are computationally expensive. As the amount of data with ever higher resolutions and frame rates increases, the problem will become even bigger. Thus the area of reserach on optimization methods for either feature extraction or action modeling constitues an open one. The ultimate goal for 3D human motion perception will be real-time analysis, requiring real-time video representation and on-line modeling, two operations that require low computational cost.

References

[1] G. Johansson. Visual motion perception. *Scientific American*, 6:76–88, 1975. 1

[2] J.A. Webb and J.K. Aggarwal. Structure from motion of rigid and jointed objects. *Artificial Intelligence*, 19, 1982. 1

[3] J.K. Aggarwal and Q Cai. Human motion analysis: A review. *Computer Vision and Image Understanding*, 3, 1999. 1

[4] L. W. Campbell and A. F. Bobick. Recognition of human body motion using phase space constraints. In *IEEE International Conference on Computer Vision*, pages 624–630, 1995. 2

[5] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1297–1304, 2011. 3, 13, 16, 29, 30, 36, 42, 45, 47

[6] Heng Wang, Muhammad Muneeb Ullah, Alexander Kläser, Ivan Laptev, and Cordelia Schmid. Evaluation of local spatio-temporal features for action recognition. In *British Machine Vision Conference*, 2009. 4, 24, 31, 32, 50

[7] J.K. Aggarwal and M.S. Ryoo. Human activity analysis: A review. *ACM Computing Surveys*, 43, 2011. 13, 30

[8] Ronald Poppe. A survey on vision-based human action recognition. *Image and Vision Computing*, 28(6):976–990, 2010. 13

[9] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea. Machine recognition of human activities: A survey. *IEEE Transactions on Circuits and Systems for Video Technology,*, 18(11):1473–1488, 2008. 13

[10] Juan Carlos Niebles, Hongcheng Wang, and Li Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. *International Journal of Computer Vision*, 79(3):299–318, 2008. 13

[11] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *IEEE International Conference on Computer Vision*, pages 1395–1402, 2005. 13

[12] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: a local svm approach. In *International Conference on Pattern Recognition*, volume 3, pages 32–36, 2004. 13

[13] Daniel Weinland, Remi Ronfard, and Edmond Boyer. Free viewpoint action recognition using motion history volumes. *Computer Vision and Image Understanding*, 104(2):249–257, 2006. 13

[14] M.D. Rodriguez, J. Ahmed, and M. Shah. Action mach a spatio-temporal maximum average correlation height filter for action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 13

[15] M. Marszalek, I. Laptev, and C. Schmid. Actions in context. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2929–2936, 2009. 13

[16] D.M Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73:82–98, 1999. 13

[17] V. Krüger, D. Kragic, A. Ude, and C. Geib. The Meaning of Action: A Review on action recognition and mapping. *Advanced Robotics*, 21(13):1473–1501, 2007. 13

[18] Thomas B. Moeslund, Adrian Hilton, and Volker Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2):90–126, 2006. 13

[19] Liang Wang, Weiming Hu, and Tieniu Tan. Recent developments in human motion analysis. *Pattern Recognition*, 36(3):585–601, 2003. 13

[20] Jiang Wang, Zicheng Liu, Ying Wu, and Junsong Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1290–1297, 2012. 15, 16, 25, 30, 55, 56, 57

[21] Long Zhu, Yuanhao Chen, Yifei Lu, Chenxi Lin, and A. Yuille. Max margin and/or graph learning for parsing the human body. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 15

[22] Bangpeng Yao and Li Fei-Fei. Grouplet: A structured image representation for recognizing human and object interactions. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9–16, 2010. 15

[23] Guang Chen, Manuel Giuliani, Daniel Clarke, and Alois Knoll. Action recognition using ensemble weighted multi-instance learning. In *IEEE International Conference on Robotics and Automation*, 2014. 15

[24] Guang Chen, Daniel Clarke, and Alois Knoll. Learning weighted joint-based features for action recognition using depth camera. In *International Conference on Computer Vision Theory and Applications*, 2014. 15, 16

[25] Francis R. Bach and Gert R. G. Lanckriet. Multiple kernel learning, conic duality, and the smo algorithm. In *International Conference on Machine Learning*, volume 69, 2004. 15

[26] L. Xia and J.K. Aggarwal. Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013. 15, 27, 50, 55, 56

[27] Omar Oreifej and Zicheng Liu. Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 716–723, 2013. vii, 15, 25, 28

[28] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy. Berkeley mhad: A comprehensive multimodal human action database. In *IEEE Workshop on Applications of Computer Vision*, pages 53–60, 2013. 16

[29] C. Keskin, F. Kirac, Y.E. Kara, and L. Akarun. Real time hand pose estimation using depth sensors. In *IEEE International Conference on Computer Vision Workshops*, pages 1228–1234, Nov 2011. 16

[30] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001. 16

[31] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986. 16

[32] Ling Gan and Fu Chen. Human action recognition using apj3d and random forests. *Journal of Software*, 8(9), 2013. 16

[33] Hossein Rehmani, Arif Mahmood, Ajmal Mian, and Du Huynh. Real time action recognition using histograms of depth gradients and random decision forests. In *IEEE Winter Applications of Computer Vision Conference*, 2014. 16

[34] Yu Zhu, Wenbin Chen, and Guodong Guo. Fusing spatiotemporal features and joints for 3d action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 486–491, 2013. 16

[35] Maxime Devanne, Hazem Wannous, Stefano Berretti, Pietro Pala, Mohamed Daoudi, and Alberto Bimbo. Space-time pose representation for 3d human action recognition. In *New Trends in Image Analysis and Processing*, volume 8158 of *Lecture Notes in Computer Science*, pages 456–464, 2013. 17

[36] Gioia Ballin, Matteo Munaro, and Emanuele Menegatti. Human action recognition from rgb-d frames based on real-time 3d optical flow estimation. In *Biologically Inspired Cognitive Architectures 2012*, volume 196 of *Advances in Intelligent Systems and Computing*, pages 65–74. 2013. 17

[37] O. Boiman, E. Shechtman, and M. Irani. In defense of nearest-neighbor based image classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 17

[38] Xiaodong Yang and Yingli Tian. Eigenjoints-based action recognition using naïve-bayes-nearest-neighbor. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 14–19, 2012. 17, 29, 55, 56

[39] Lorenzo Seidenari, Vincenzo Varano, Stefano Berretti, Pietro Pala, and Alberto Del Bimbo. Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses. In *IEEE Workshop on Human Activity Understanding from 3D Data*, 2013. 17, 30

[40] A. Veeraraghavan, A.K. Roy-Chowdhury, and R. Chellappa. Matching shape sequences in video with applications in human movement analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12):1896–1909, 2005. 18

[41] A. Veeraraghavan, R. Chellappa, and A.K. Roy-Chowdhury. The function space of an activity. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 959–968, 2006. 18

[42] Alexandros Andre Chaaraoui, Jose Ramon Padilla-Lopez, and Francisco Florez-Revuelta. Fusion of skeletal and silhouette-based features for human action recognition with rgb-d devices. In *IEEE Workshop on Consumer Depth Cameras for Computer Vision*, December 2013. 18

[43] Alexandros Andre Chaaraoui, JosRamón Padilla-López, Pau Climent-Pérez, and Francisco Flórez-Revuelta. Evolutionary joint selection to improve human action recognition with rgb-d devices. *Expert Systems with Applications*, 41:786–794, 2014. 18

[44] M. Reyes, G. Dominguez, and S. Escalera. Feature weighting in dynamic time warping for gesture recognition in depth data. In *IEEE International Conference on Computer Vision Workshops*, pages 1182–1188, 2011. 18

[45] A. Hernandez-Vela, M.A. Bautista, X. Perez-Sala, V. Ponce, X. Baro, O. Pujol, C. Angulo, and S. Escalera. Bovdw: Bag-of-visual-and-depth-words for gesture recognition. In *International Conference on Pattern Recognition*, pages 449–452, 2012. 18, 25, 27

[46] MiguelÁngel Bautista, Antonio Hernández-Vela, Victor Ponce, Xavier Perez-Sala, Bar Xavier, Oriol Pujol, Cecilio Angulo, and Sergio Escalera. Probability-based dynamic time warping for gesture recognition on rgb-d data. In *Advances in Depth Image Analysis and Applications*, volume 7854 of *Lecture Notes in Computer Science*, pages 126–135. 2013. 18

[47] Jiang Wang and Ying Wu. Learning maximum margin temporal warping for action recognition. In *The IEEE International Conference on Computer Vision*, 2013. 18

[48] J. Yamato, Jun Ohya, and K. Ishii. Recognizing human action in time-sequential images using hidden markov model. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 379–385, 1992. 18

[49] Xiaolin Feng and P. Perona. Human action recognition by sequence of movelet codewords. In *International Symposium on 3D Data Processing Visualization and Transmission*, pages 717–721, 2002. 18

[50] Fengjun Lv and R. Nevatia. Single view human action recognition using key pose matching and viterbi path searching. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007. 18

[51] L. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989. 19

[52] Andre Gaschler, Sören Jentzsch, Manuel Giuliani, Kerstin Huth, Jan de Ruiter, and Alois Knoll. Social Behavior Recognition using body posture and head pose for Human-Robot Interaction. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012. 19

[53] Andre Gaschler, Kerstin Huth, Manuel Giuliani, Ingmar Kessler, Jan de Ruiter, and Alois Knoll. Modelling state of interaction from head poses for social Human-Robot Interaction. In *ACM/IEEE HCI Conference on Gaze in Human-Robot Interaction Workshop*, 2012. 19

[54] Lu Xia, Chia-Chih Chen, and J.K. Aggarwal. View invariant human action recognition using histograms of 3d joints. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 20–27, 2012. 19, 29

[55] A. Dubois and F. Charpillet. Human activities recognition with rgb-depth camera using hmm. In *IEEE Conference on Engineering in Medicine and Biology Society*, pages 4666–4669, 2013. 19

[56] GeorgiosTh. Papadopoulos, Apostolos Axenopoulos, and Petros Daras. Real-time skeleton-tracking-based human action recognition using kinect data. In *MultiMedia Modeling*, volume 8325 of *Lecture Notes in Computer Science*, pages 473–483. 2014. 19

[57] Dimitrios I. Kosmopoulos, Paul Doliotis, and Ilias Maglogiannis. Fusion of color and depth video for human behavior recognition in an assistive environment. In *International Conference on Human-Computer Interaction*, pages 42–51, 2013. 19

[58] Wanqing Li, Zhengyou Zhang, and Zicheng Liu. Expandable data-driven graphical modeling of human actions based on salient postures. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(11):1499–1510, 2008. 19, 20

[59] AntonioW. Vieira, EricksonR. Nascimento, GabrielL. Oliveira, Zicheng Liu, and MarioF.M. Campos. Stop: Space-time occupancy patterns for 3d action recognition from depth map sequences. In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, volume 7441 of *Lecture Notes in Computer Science*, pages 252–259. 2012. 19, 20, 28

[60] A. Kurakin, Z. Zhang, and Z. Liu. A real time system for dynamic hand gesture recognition with a depth sensor. In *European Signal Processing Conference*, pages 1975–1979, 2012. 19

[61] E. Fox, E.B. Sudderth, M.I. Jordan, and A. Willsky. Bayesian nonparametric inference of switching dynamic linear models. *IEEE Transactions on Signal Processing*, 59(4): 1569–1585, 2011. 19

[62] SangMin Oh, JamesM. Rehg, Tucker Balch, and Frank Dellaert. Learning and inferring motion patterns using parametric segmental switching linear dynamic systems. *International Journal of Computer Vision*, 77(1-3):103–124, 2008. 19

[63] A. Bargi, R.Y.D. Xu, and M. Piccardi. An online hdp-hmm for joint action segmentation and classification in motion capture data. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–7, 2012. 19

[64] C. Zhang and Y. Tian. Rgb-d camera-based daily living activity recognition. *International Journal of Computer Vision and Image Processing*, 2, 2012. 20

[65] Chenyang Zhang and Yingli Tian. Rgb-d camera-based activity analysis. In *Signal Information Processing Association Annual Summit and Conference*, pages 1–6, 2012. 20

[66] Chenyang Zhang, Yingli Tian, and Elizabeth Capezuti. Privacy preserving automatic fall detection for elderly using rgbd cameras. In *International conference on Computers Helping People with Special Needs*, pages 625–633, 2012. 20

[67] Jaeyong Sung, Colin Ponce, Bart Selman, and Ashutosh Saxena. Human activity detection from rgbd images. In *AAAI Workshops on Plan, Activity, and Intent Recognition*, 2011. 20

[68] Jaeyong Sung, C. Ponce, B. Selman, and A. Saxena. Unstructured human activity detection from rgbd images. In *IEEE International Conference on Robotics and Automation*, pages 842–849, 2012. 20

[69] Andrew McCallum, Dayne Freitag, and Fernando Pereira. Maximum entropy markov models for information extraction and segmentation. In *International Conference on Machine Learning*, pages 591–598, 2000. 20

[70] Bingbing Ni, Yong Pei, P. Moulin, and Shuicheng Yan. Multilevel depth and image fusion for human activity detection. *IEEE Transactions on System, Man and Cybernetics (B)*, 43(5):1383–1394, 2013. 20

[71] Bingbing Ni, Yong Pei, Zhujin Liang, Liang Lin, and P. Moulin. Integrating multi-stage depth-induced contextual information for human action recognition and localization. In *IEEE International Conference on Automatic Face and Gesture Recognition*, pages 1–8, 2013. 20

[72] Chun-Nam John Yu and Thorsten Joachims. Learning structural svms with latent variables. In *International Conference on Machine Learning*, pages 1169–1176, 2009. 20

[73] Yang Wang and G. Mori. Hidden part models for human action recognition: Probabilistic versus max margin. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(7):1310–1323, 2011. 20

[74] Hema Koppula and Ashutosh Saxena. Anticipating human activities using object affordances for reactive robotic response. In *Robotics: Science and Systems Conference*, 2013. 20

[75] Hema Koppula, Rudhir Gupta, and Ashutosh Saxena. Learning human activities and object affordances from rgb-d videos. *International Journal of Robotics Research*, 32(8): 951–970, 2013. vii, 20, 21

[76] Hema Koppula and Ashutosh Saxena. Learning spatio-temporal structure from rgb-d videos for human activity detection and anticipation. In *International Conference on Machine Learning*, volume 28, pages 792–800, 2013. 20

[77] Kris M. Kitani, Brian D. Ziebart, James Andrew Bagnell, and Martial Hebert. Activity forecasting. In *European Conference on Computer Vision*, volume 7575 of *Lecture Notes in Computer Science*, pages 201–214, 2012. 20

[78] Ivan Laptev. On space-time interest points. *International Journal of Computer Vision*, 64(2-3):107–123, 2005. 23

[79] Honglak Lee, Roger Grosse, Rajesh Ranganath, and Andrew Y. Ng. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *International Conference on Machine Learning*, pages 609–616, 2009. 23, 33

[80] Guillaume Desjardins and Yoshua Bengio. Empirical evaluation of convolutional RBMs for vision. Technical Report 1327, Département d'Informatique et de Recherche Opérationnelle, Université de Montréal, 2008. 23, 33

[81] Bo Chen, Jo-Anne Ting, Benjamin Marlin, and Nando de Freitas. Deep learning of invariant spatio-temporal features from video. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*. 2010. 23, 33

[82] Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Y. Ng. Efficient sparse coding algorithms. In *Advances in Neural Information Processing Systems 20*, pages 801–808, 2007. 23

[83] Jiquan Ngiam, Pang Wei Koh, Zhenghao Chen, Sonia A. Bhaskar, and Andrew Y. Ng. Sparse filtering. In *Advances in Neural Information Processing Systems 24*, pages 1125–1133, 2011. 23

[84] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):221–231, 2013. 24, 33

[85] Q.V. Le, W.Y. Zou, S.Y. Yeung, and A.Y. Ng. Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3361–3368, 2011. 24, 32, 33, 34, 68, 73

[86] Kevin Lai, Liefeng Bo, Xiaofeng Ren, and Dieter Fox. Rgb-d object recognition: Features, algorithms, and a large scale benchmark. In *Consumer Depth Cameras for Computer Vision: Research Topics and Applications*, pages 167–192. Springer, 2013. 24

[87] Li Deng, G. Hinton, and B. Kingsbury. New types of deep neural network learning for speech recognition and related applications: an overview. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 8599–8603, 2013. 24

[88] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D. Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, Xin Zhang, Jake Zhao, and Karol Zieba. End to end learning for self-driving cars. *CoRR*, 2016. 24

[89] I. Laptev and T. Lindeberg. Space-time interest points. In *IEEE International Conference on Computer Vision*, pages 432–439, 2003. 24, 25, 31, 39

[90] Geert Willems, Tinne Tuytelaars, and Luc J. Van Gool. An efficient dense and scale-invariant spatio-temporal interest point detector. In *European Conference on Computer Vision*, volume 5303 of *Lecture Notes in Computer Science*, pages 650–663, 2008. 24, 25

[91] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *IEEE Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pages 65–72, 2005. 24, 25

[92] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 886–893, 2005. 24, 25, 26, 28, 31, 39

[93] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 24, 26

[94] Bingbing Ni, Gang Wang, and P. Moulin. Rgbd-hudaact: A color-depth video database for human daily activity recognition. In *IEEE International Conference on Computer Vision Workshops*, pages 1147–1153, 2011. 25, 26

[95] Simon Hadfield and Richard Bowden. Hollywood 3d: Recognizing actions in 3d natural scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3398–3405, 2013. 25, 26

[96] Yang Zhao, Zicheng Liu, Lu Yang, and Hong Cheng. Combing rgb and depth map features for human activity recognition. In *Signal Information Processing Association Annual Summit and Conference*, pages 1–4, 2012. 26, 27

[97] Olusegun Oshin, Andrew Gilbert, and Richard Bowden. Capturing the relative distribution of features for action recognition. In *IEEE International Conference on Automatic Face & Gesture Recognition and Workshops*, pages 111–116, 2011. 26

[98] A.F. Bobick and J.W. Davis. The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(3):257–267, 2001. 26

[99] Xiaodong Yang, Chenyang Zhang, and YingLi Tian. Recognizing actions using depth motion maps-based histograms of oriented gradients. In *ACM international conference on Multimedia*, pages 1057–1060, 2012. 26

[100] Torsten Fiolka, Jörg Stückler, DominikA. Klein, Dirk Schulz, and Sven Behnke. Sure: Surface entropy for distinctive 3d features. In *Spatial Cognition VIII*, volume 7463 of *Lecture Notes in Computer Science*, pages 74–93. 2012. 26

[101] B. Steder, R. B. Rusu, K. Konolige, and W. Burgard. NARF: 3D range image features for object recognition. In *IEEE Workshop on Defining and Solving Realistic Perception Problems in Personal Robotics*, 2010. 26

[102] R.B. Rusu, G. Bradski, R. Thibaux, and J. Hsu. Fast 3d recognition and pose using the viewpoint feature histogram. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2155–2162, 2010. 26, 27

[103] R.B. Rusu, N. Blodow, and M. Beetz. Fast point feature histograms (fpfh) for 3d registration. In *IEEE International Conference on Robotics and Automation*, pages 3212–3217, 2009. 26

[104] Zhongwei Cheng, Lei Qin, Yituo Ye, Qingming Huang, and Qi Tian. Human daily action analysis with multi-view and color-depth data. In *ECCV Workshops and Demonstrations*, volume 7584 of *Lecture Notes in Computer Science*, pages 52–61, 2012. 27

[105] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12):2037–2041, 2006. 27

[106] Jiang Wang, Zicheng Liu, Jan Chorowski, Zhuoyuan Chen, and Ying Wu. Robust 3d action recognition with random occupancy patterns. In *European conference on Computer Vision*, pages 872–885, 2012. 27, 55, 56

[107] E. Shechtman and M. Irani. Matching local self-similarities across images and videos. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007. 27

[108] A.A. Efros, A.C. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *IEEE International Conference on Computer Vision*, pages 726–733, 2003. 28

[109] P.A. Crook, V. Kellokumpu, G. Zhao, and M. Pietikainen. Human activity recognition using a dynamic texture based method. In *British Machine Vision Conference*, 2008. 28

[110] Wei-Lwun Lu and J.J. Little. Simultaneous tracking and action recognition using the pca-hog descriptor. In *Canadian Conference on Computer and Robot Vision*, pages 6–14, 2006. 28

[111] N. Ikizler, R.G. Cinbis, S. Pehlivan, and P. Duygulu. Recognizing actions from still images. In *International Conference on Pattern Recognition*, pages 1–4, 2008. 28

[112] Somayeh Danafar and Niloofar Gheissari. Action recognition for surveillance applications using optic flow and svm. In *Asia Conference on Computer Vision*, volume 4844 of *Lecture Notes in Computer Science*, pages 457–466. 2007. 28

[113] A. Klaeser, M. Marszalek, and C. Schmid. A spatio-temporal descriptor based on 3d-gradients. In *British Machine Vision Conference*, pages 99.1–99.10, 2008. 28

[114] Shuai Tang, Xiaoyu Wang, Xutao Lv, TonyX. Han, James Keller, Zhihai He, Marjorie Skubic, and Shihong Lao. Histogram of oriented normal vectors for object recognition with a depth sensor. In *Asian Conference on Computer Vision*, volume 7725 of *Lecture Notes in Computer Science*, pages 525–538. 2013. 28

[115] Gunnar Johansson. *Visual Motion Perception*, volume 564. Scientific American, 1975. 28, 30

[116] Gunnar Johansson. Visual perception of biological motion and a model for its analysis. *Perception & Psychophysics*, 14:201–211, 1973. 28

[117] Jon A. Webb and J.K. Aggarwal. Structure from motion of rigid and jointed objects. *Artificial Intelligence*, 19(1):107–130, 1982. 28

[118] Sourabh A. Niyogi and Edward H. Adelson. Analyzing and recognizing walking figures in xyt. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 469–474, 1994. 28

[119] L.W. Campbell and A.F. Bobick. Recognition of human body motion using phase space constraints. In *Internaltional Conference on Computer Vision*, pages 624–630, 1995. 28

[120] Yaser Sheikh, Mumtaz Sheikh, and Mubarak Shah. Exploring the space of a human action. In *IEEE International Conference on Computer Vision*, pages 144–149, 2005. 28

[121] A. Yilma and M. Shah. Recognizing human actions in videos acquired by uncalibrated moving cameras. In *International Conference on Computer Vision*, pages 150–157, 2005. 28

[122] Meinard Müller, Andreas Baak, and Hans-Peter Seidel. Efficient and robust annotation of motion capture data. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 17–26, 2009. 29

[123] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, and A. Weber. Documentation mocap database hdm05. Technical Report CG-2007-2, Universität Bonn, 2007. 29

[124] Lei Han, Xinxiao Wu, Wei Liang, Guangming Hou, and Yunde Jia. Discriminative human action recognition in the learned hierarchical manifold space. *Image and Vision Computing*, 28(5):836–849, 2010. 29

[125] Nikolaos Gkalelis, Hansung Kim, Adrian Hilton, Nikos Nikolaidis, and Ioannis Pitas. The i3dpost multi-view and 3d human action/interaction database. In *IEEE Conference for Visual Media Production*, pages 159–168, 2009. 29

[126] H. Ragheb, S. Velastin, P. Remagnino, and T. Ellis. ViHASi: Virtual human action silhouette data for the performance evaluation of silhouette-based action recognition methods. In *ACM/IEEE International Conference on Distributed Smart Cameras*, pages 1–10, 2008. 29

[127] D. Weinland, E. Boyer, and R. Ronfard. Action recognition from arbitrary views using 3d exemplars. In *IEEE International Conference on Computer Vision*, pages 1–7, 2007. 29

[128] R. Girshick, J. Shotton, P. Kohli, Antonio Criminisi, and A. Fitzgibbon. Efficient regression of general-activity human poses from depth images. In *IEEE International Conference on Computer Vision*, pages 415–422, 2011. 29

[129] Jamie Shotton, Ross Girshick, Andrew Fitzgibbon, Toby Sharp, Mat Cook, Mark Finocchio, Richard Moore, Pushmeet Kohli, Antonio Criminisi, Alex Kipman, and Andrew

Blake. Efficient human pose estimation from single depth images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2821–2840, 2013. 29

[130] Min Sun, P. Kohli, and J. Shotton. Conditional regression forests for human pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3394–3401, 2012. 29

[131] Xi Chen and Markus Koskela. Skeleton-based action recognition with extreme learning machines. In *Neurocomputing*, 2013. 29

[132] Xiaodong Yang and YingLi Tian. Effective 3d action recognition using eigenjoints. *Journal of Visual Communication and Image Representation*, 2014. 29

[133] V. Bloom, D. Makris, and V. Argyriou. G3d: A gaming action dataset and real time action recognition evaluation framework. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 7–12, 2012. 29

[134] Kiwon Yun, Jean Honorio, Debaleena Chattopadhyay, Tamara L. Berg, and Dimitris Samaras. Two-person interaction detection using body-pose features and multiple instance learning. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 28–35, 2012. 29

[135] Meinard Müller, Tido Röder, and Michael Clausen. Efficient content-based retrieval of motion capture data. *ACM Transactions on Graphics*, 24(3):677–685, 2005. 29

[136] Angela Yao, Juergen Gall, Gabriele Fanelli, and Luc Van Gool. Does human action recognition benefit from pose estimation? In *British Machine Vision Conference*, pages 67.1–67.11, 2011. 29, 31

[137] Yukako Yamane, Eric T Carlson, Katherine C Bowman, Zhihong Wang, and Charles E Connor. A neural code for three-dimensional object shape in macaque inferotemporal cortex. *Nature Neuroscience*, 11(11):1352–1360, 2008. 29

[138] Chia-Chun Hung, Eric T. Carlson, and Charles E. Connor. Medial axis shape coding in macaque inferotemporal cortex. *Neuron*, 74(6):1099–1113, 2012. 29

[139] Rizwan Chaudhry, Ferda Ofli, Gregorij Kurillo, Ruzena Bajcsy, and RenVidal. Bio-inspired dynamic 3d discriminative skeletal features for human action recognition. In *IEEE Workshop on Human Activity Understanding from 3D Data*, 2013. 29

[140] Liang Wang, Yizhou Wang, Tingting Jiang, and Wen Gao. Instantly telling what happens in a video sequence using simple features. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3257–3264, 2011. 30

[141] Chunyu Wang, Yizhou Wang, and A.L. Yuille. An approach to pose-based action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 915–922, 2013. vii, 30, 55, 56

[142] Guozhu Dong and Jinyan Li. Efficient mining of emerging patterns: Discovering trends and differences. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 43–52, 1999. 30

[143] Michalis Raptis, Darko Kirovski, and Hugues Hoppe. Real-time classification of dance gestures from skeleton animation. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 147–156, 2011. 30

[144] Mohammad A. Gowayyed, Marwan Torki, Mohamed E. Hussein, and Motaz El-Saban. Histogram of oriented displacements (HOD): Describing trajectories of human joints for action recognition. In *International Joint Conference on Artificial Intelligence*, pages 1351–1357, 2013. 31, 55, 56, 57

[145] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy. Sequence of the most informative joints (smij): A new representation for human skeletal action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 8–13, 2012. 31

[146] Oncel Tuzel, Fatih Porikli, and Peter Meer. Region covariance: A fast descriptor for detection and classification. In *European Conference on Computer Vision*, pages 589–600, 2006. 31

[147] O. Tuzel, F. Porikli, and P. Meer. Pedestrian detection via classification on riemannian manifolds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(10): 1713–1727, 2008. 31

[148] A. Sanin, C. Sanderson, M.T. Harandi, and B.C. Lovell. Spatio-temporal covariance descriptors for action and gesture recognition. In *IEEE Workshop on Applications of Computer Vision*, pages 103–110, 2013. 31

[149] Mohamed E. Hussein, Marwan Torki, Mohammad A. Gowayyed, and Motaz El-Saban. Human action recognition using a temporal hierarchy of covariance descriptors on 3d joint locations. In *International Joint Conference on Artificial Intelligence*, pages 2466–2472, 2013. 31

[150] Geoffrey E. Hinton, Simon Osindero, and Yee Whye Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554, 2006. 32, 34

[151] M. Norouzi, M. Ranjbar, and G. Mori. Stacks of convolutional restricted boltzmann machines for shift-invariant feature learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2735–2742, 2009. 33

[152] Aapo Hyvrinen, Jarmo Hurri, and Patrick O. Hoyer. *Natural Image Statistics: A Probabilistic Approach to Early Computational Vision.* Springer Publishing Company, Incorporated, 1st edition, 2009. 33

[153] Zhi-Hua Zhou. Ensemble learning. In *Encyclopedia of Biometrics*, pages 270–273, 2009. 40

[154] Wanqing Li, Zhengyou Zhang, and Zicheng Liu. Action recognition based on a bag of 3d points. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 9–14, 2010. 40, 50, 56

[155] J.C.A. Read, G.P. Phillipson, I. Serrano-Pedraza, A.D. Milner, and A.J. Parker. Stereoscopic vision in the absence of the lateral occipital cortex. *PLoS One*, 5, 2010. 42

[156] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *IEEE Conferenceon Computer Vision and Pattern Recognition*, volume 2, pages 2169–2178, 2006. 43

[157] Xu-Ying Liu, Jianxin Wu, and Zhi-Hua Zhou. Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 39(2):539–550, 2009. 47

[158] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55:119–139, 1997. 47

[159] S. V. N. Vishwanathan, Z. Sun, N. Theera-Ampornpunt, and M. Varma. Multiple kernel learning and the SMO algorithm. In *Advances in Neural Information Processing Systems 23*, 2010. 48

[160] John C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74, 1999. 49

[161] Wanqing Li, Zhengyou Zhang, and Zicheng Liu. Action recognition based on a bag of 3d points. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 9–14, 2010. 50, 55

[162] Omar Oreifej and Zicheng Liu. Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 716–723, 2013. 50, 55, 56

[163] A. Bosch, A. Zisserman, and X. Muoz. Image classification using random forests and ferns. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8, Oct 2007. 69, 79

[164] Bangpeng Yao, Aditya Khosla, and Li Fei-Fei. Combining randomization and discrimination for fine-grained image categorization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2011. 73, 77

[165] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001. 77

[166] Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, T. Huang, and Yihong Gong. Locality-constrained linear coding for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3360–3367, June 2010. 80

[167] Vladimir I. Pavlovic, Rajeev Sharma, and Thomas S. Huang. Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(7):677–695, 1997. 85