8-1-2013

Human Action Recognition via Fused Kinematic Structure and Surface Representation

Salah R. Althloothi
University of Denver

Recommended Citation

Althloothi, Salah R., "Human Action Recognition via Fused Kinematic Structure and Surface Representation" (2013). *Electronic Theses and Dissertations*. 27.
https://digitalcommons.du.edu/etd/27

Human Action Recognition Via Fused Kinematic Structure and Surface Representation

A Dissertation
Presented to
the Faculty of the Daniel Felix Ritchie
School of Engineering and Computer Science
University of Denver

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

by
Salah R. Althloothi

August 2013

Advisor: Dr. Mohammad H. Mahoor

Author: Salah R. Althloothi
Title: Human Action Recognition Via Fused Kinematic Structure and Surface Representation
Advisor: Dr. Mohammad H. Mahoor
Degree Date: August 2013

Abstract

Human action recognition from visual data has remained a challenging problem in the field of computer vision and pattern recognition. This dissertation introduces a new methodology for human action recognition using motion features extracted from kinematic structure and shape features extracted from surface representation of the human body. Motion features are used to provide sufficient information about human movement, whereas shape features are used to describe the structure of silhouette. These features are fused at the kernel level using Multikernel Learning (MKL) technique to enhance the overall performance of human action recognition. In fact, there are advantages in using multiple types of features for human action recognition, especially if the features are complementary to each other (e.g. kinematic/motion features and shape features). For instance, challenging problems such as inter-class similarity among actions and performance variation, which cannot be resolved easily by using a single type of feature, can be handled by fusing multiple types of features.

This dissertation presents a new method for representing the human body surface provided by depth map (3-D) using spherical harmonics representation. The advantage of using the spherical harmonics representation is to represent the whole body surface into a finite series of spherical harmonics coefficients. Furthermore, these series can be used to describe the pose of the body using the phase information encoded inside the coefficients. Another method for detecting/tracking distal limb segments using the kinematic structure is developed. The advantage of using the distal limb segments is to extract discriminative features that can provide sufficient and compact information to recognize human actions. Our experimental results show that the aforementioned methods for human action description are complementary to each other. Hence, combining both features can enhance the robustness of action recognition. In this context, a framework to fuse multiple features using the MKL technique is developed. The experimental results show that this framework is promising in incorporating multiple features in different domains for automated recognition of human actions.

Table of Contents

Chapter 1

Chapter 2

Chapter 3

List of Figures

List of Tables

List of Abbreviations

DT: Distance Transform

DMM: Depth Motion Maps

EM: Expectation-Maximization

GMM: Gaussian Mixture Model

HOG: Histograms of Oriented Gradients

IMU: Inertial Measurement Unit

MAT: Medial Axis Transform

LOSO: Leave One Subject Out

MEI: Motion Energy Images

MEMM: Maximum Entropy Markov Model

MHI: Motion History Images

MKL: MultiKernel Learning

MSE: Mean Square Error

MSR: Microsoft Research

PCA: Principal Component Analysis

ROI: Region Of Interest

SDK: Software Development Kit

SIFT: Scale Invariant Feature Transform

SURF: Speeded-Up Robust Feature

Acknowledgements

First and foremost, I would like to express my sincere gratitude to Dr. Mohammad Mahoor, for his valuable support, encouragement and guidance. He introduced me to the field of computer vision, and helped me to jump start my research career in this field. My thanks also go to Dr. Richard Voyles, for his valuable comments and encouragement during my Ph.D study. Despite his busy schedule, he would always set aside plenty of time to discuss some concepts related to the research with me and patiently monitor my attempts to do research. Both of them encouraged me to pursue novel ideas and provided me with exceptional experience and knowledge. My thanks also go to the committee member, Dr. Jun Jason Zhang, and Dr. Nikolaos Galatos for their comments and suggestions.

In addition, I would like to extend my thanks to the faculty and staff members of the Electrical and Computer Engineering Department at the University of Denver. Also, I would like to thank my colleagues at the Computer Vision and Robotics Lab, specially Robert Nawrocki and Mohammad Mavadati, for their support and help during my study at the University of Denver. Finally, I would like to thank my fellow lab members for their collaboration during the completion of this thesis.

Chapter 1

Introduction

Human action recognition from visual data has been an active research topic in computer vision and pattern recognition. Particularly, automated recognition of human activities from sequence of images has been one of the most active research areas in this field. There are many promising applications such as automated visual surveillance, human-robot interaction, video retrieval, and motion-based human identification for automated human action recognition.

In the past decades, significant amount of research has been done in the area of human action recognition using either shape features or motion features extracted from sequence of 2-D images. According to the survey of Moeslund et al. [8], human movement can be divided into three categories: limb action, whole-body action, and activity. The limb action is a movement that can be described at the limb level. The whole-body action consists of a set of human limbs movement that can be described at the whole-body movement for a short period. The activity contains a number of subsequent whole-body actions in a cycle which gives an interpretation of the movement that is being performed in this cycle. For example, ”moving the right leg forward” is a limb action, and ”walking” is the whole-body action because it consists of a set of limbs actions. While, ”fighting” is an activity that contains a number of subsequent actions such as standing, jumping, running, etc. In this dissertation, we adopt this classification in recognizing limb actions and whole-body actions.

Based on Moeslund et al. [8] classification, limb actions are classified as primitive actions that can be used to describe complex actions (i.e., whole-body action and activity). One of the challenges of detecting/recognizing limb actions is to accurately detect human body limbs and then describe body motion using the detected limbs as Poppe [9] reported in his article.

Most of the early developed methods for human action recognition are mainly concentrated on sequence of 2-D images captured by regular RGB cameras. This is due to the fact that the surveillance and security systems utilized regular 2-D cameras for video acquisition. Recently, developed depth sensors such as Kinect have opened up new possibilities of dealing with 3-D data. Thus, a great attention is being paid to extract 3-D spatio-temporal features based on sequences of depth information using Kinect sensor to describe human action. The idea behind using 3-D spatio-temporal features is to detect/describe the changes in the body shape based on dominant motions in body limbs. These features mainly focus on the representation of space-time shapes. Consequently, the main concept is to recognize human action by detecting/describing the changes in the body limbs either by detecting the 3-D motion of body limbs or through measuring similarities among different 3-D space-time shapes.

In the last two years, the research in human action recognition based on sequence of depth maps has motivated with the release of the Kinect (RGBD sensor) for windows SDK [10], which provides the estimation of the 3-D joint positions. Although the RGBD sensors produce a better quality of estimating 3-D joint positions than those estimated from RGB sensors, the estimated 3-D joint positions are still noisy and fail when there are occlusions between human limbs such as two limbs crossing each other. In addition, the 3-D joint positions alone are insufficient to distinguish similar actions such as ”drinking” and ”eating”, as illustrated

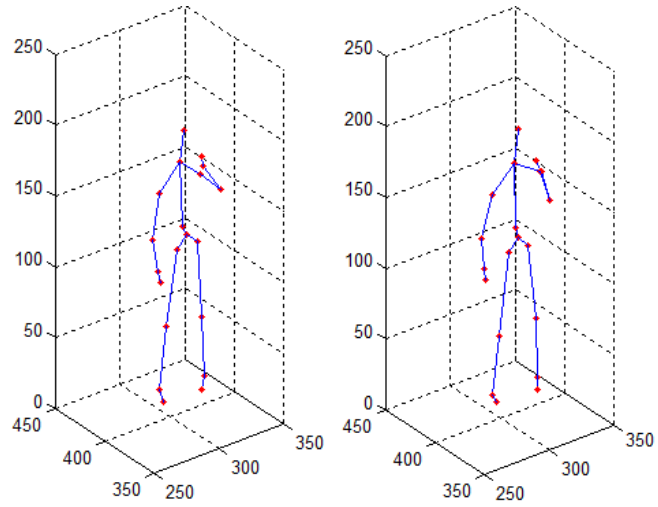


Figure 1.1. The 3-D joint positions for two similar actions ”drinking” and ”eating”

in Figures 1.1. Therefore, extra informations need to be included in the feature level to enhance the recognition rate. In this context, we developed a method that can be used to fuse multiple types of features in order to boost the recognition rate for human action recognition. Hence, this dissertation presents two sets of features based on surface representation and kinematic structure for human action recognition. These two sets of feature are fused using MKL technique at the kernel level to enhance the recognition rate for human actions. The advantage of using the MKL is to improve the generalization performance since a single kernel may not perform well for different types of features.

1.1 Motivation

In many surveillance situations such as access control and parking lot monitoring, it is necessary to not only know if humans are present but more importantly what they are doing. Therefore, the research topic is motivated by many applications such as surveillance systems, video analysis, human-robot interaction and a variety of systems that involve interactions between human and computerized machines such as human-computer interfaces.

In general, human action recognition has a wide range of potential applications in our life. Some of these potential applications are summarized in the following points:

- *Visual surveillance* in many security areas, such as banks, department stores, parking lots, and borders, have strong need for automated surveillance systems.

- *Computer user interface* is an advanced user interface in which human motion analysis is used to interpret the control command.

- *Abnormal behaviour detection* used to analyze the behaviours of people and determine whether their behaviours are normal or abnormal.

- *Computer games* to play video games without controller devices.

- *Human motion analysis* has attracted much attention of researchers in the medical field in order to help elder people.

- *Sign-language translation* to help handicapped people.

In addition, the need to improve automated visual surveillance using a monocular view has attracted the attention of many researchers. Recent advances in imaging technology, such as Kinect sensor, have made it possible to capture real 3-D data and inexpensively using one sensor. This technology has attracted the

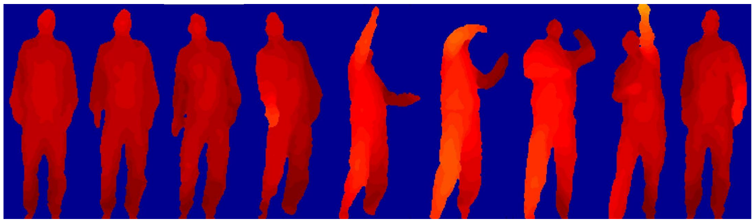


Figure 1.2. A sequence of depth maps for Tennis Serve action.

attention of many researchers, and given the computer vision community the motivation to use real 3-D information of the scene in human action recognition field. It has given the computer vision community the opportunity to acquire depth maps with a good resolution 320×240 pixels. As we can see in Figure 1.2, the depth map provides additional information as 3-D data which is expected to be helpful in distinguishing poses of silhouettes. Furthermore, human action recognition based on 3-D spatio-temporal features find more applications in various fields with the increase in the computational power of modern computers. Therefore, with the availability of RGBD sensors and modern computers, most of the recent research are motivated to utilize the depth maps as 3-D data source to design a robust 3-D spatio-temporal features.

Another motivation is that the depth map is robust to the change in color, texture and illumination. Hence, researchers are motivated to utilize 3-D points of the depth map to recognize human actions. These 3-D points can be used to describe the surface of human body shape as set of 3-D features. In this dissertation, human body surface provided by depth map is utilized to design a robust 3-D shape features to describe human action. These 3-D features are typically more invariant to illumination and texture compared with other features (e.g., SIFT, SURF) because of depth map properties.

1.2 Challenges

Automatic human action recognition is still one of a challenging task in the field of computer vision due to variations in several factors such as performance, environment and view angles variations. Furthermore, segmentation problems cause action recognition algorithms fail to recognize the correct action because human action recognition algorithms rely on segmenting the whole body.

In general, some of these challenges that influence human action recognition still need further research are summarized in the following points. For further detail, we refer our readers to the surveys by Poppe [9] and Moeslund et al. [8].

1.2.1 Performance variations

There are variations in performing actions among most people. Each actor performs an action in his/her own style and in different rate compared with other actors. For example, Waving action is different in stride length and rate among actors. However, a robust method for human action recognition should be able to generalize performance variations among one action and distinguish among actions for different actors. Additionally, recognizing a large number of actions is more challenging because of increased similarity among the actions.

1.2.2 Environment variations

The most source of variation is the environment variation where the action performance takes place. It is very challenging to detect the location (ROI) of the person in cluttered or dynamic environments. Moreover, lighting conditions or illumination of the environment can further influence the appearance of the moving person. While, a static environment less challenging than other environments to detect the location (ROI) of the person and represent the motion of the actor.

1.2.3 View-angle variations

Observing the same action from different view angles can lead to different image observations and thus to a different perceived action. In this sense, multiple cameras can alleviate this issue and a occlusion issue by combining and synchronizing all image observations from different views into a consistent scene. But, because of the synchronization problem, the challenge here becomes harder than using one camera to capture the action. Moreover, moving the camera increase the complexity of localizing the moving actor in the scene, especially when the backgrounds are dynamic. All these challenges should be addressed explicitly or fixed before the action take place in order to recognize human action.

1.2.4 Temporal variations

An important effect on the temporal variations is the rate of performance while the action is recorded, particularly when the features are extracted to represent the action. Because of the rate variations for an action, it will be more challenging to know the beginning and the ending of the action. Consequently, a robust method for human action recognition should be invariant to different rates of execution and be able to generalize rate variations among different actors.

1.3 Dissertation objectives and contributions

By considering the above challenges and issues for human action recognition, we propose a framework that can generalize the intra-class variability among actors. Furthermore, the proposed framework can distinguish the inter-class similarity among actions in order to enhance the recognition rate of action classification.

The goal of this dissertation is to introduce a new methodology for human action recognition using kinematic structure and surface representation of human body. The first scenario of our work is based on kinematic structure of human body to extract motion features which are employed for action classification. For this, existing approaches to describe motion features in videos are investigated and new features based on kinematic structure are developed. The motion features that can represent the motion of the distal limb segments are developed. These features are used to describe the motion of human body in the spatial domain. The idea behind using the distal limb segments is to convert the kinematic structure into a discriminative features that can describe the motion of human body. The second scenario of this work introduces a new method based on the surface representation of human body to extract shape features which describe structure of the silhouette. The shape features are used to describe the human body shape in the frequency domain using spherical harmonics representation. These two sets of features are fused at the kernel level to enhance the overall performance of human action recognition using MKL technique. We summarize the main contributions of this dissertation as follows:

- Reduction of the kinematic structure to the distal limbs only for action recognition. Our key observation is that the change in positions of the end points of distal limb segments can be utilized to represent the human movement as motion features.

- 3-D spatio-temporal features based on spherical harmonics representation for human action recognition is presented. By representing the depth map of the silhouette in the spherical harmonics domain, the spherical harmonics coefficients can describe the structure of the human body as a set of shape features.

- Introducing a framework that can integrate multiple features in different domains for recognizing human action. These different type of features are fused in one kernel matrix using MKL technique in order to enhance the classification performance.

- The proposed framework has the ability to describe the human-object interactions using kinematic structure and surface representation of human body. Also, the proposed framework is evaluated on two challenging datasets: MSR-Action 3-D dataset [11] and Daily Activity 3-D dataset [12]. Our experimental results show a promising direction in incorporating multiple features to enhance the classification performance.

1.4 Dissertation outline

The remainder of this dissertation is organized as follows: In Chapter two, we present a review of related work for human action recognition in two dimensional (2-D) based methods and three dimensional (3-D) based methods. Chapter three explains a general overview of the different techniques used in fusion. Particularly, a feature level fusion for the task of human action recognition using MKL technique is presented in more detail. Chapter four presents a 2-D based approach for detecting/tracking distal limb segments using kinematic structure of human body created from visual data. Furthermore, the motion features for human action recognition with the experimental results are presented. Chapter five presents a 3-D based approach for human action recognition based on the surface of human body using spherical harmonics representation with the experimental results. Chapter six presents our multi-modal system for human action recognition using MKL technique. We provide detailed procedures of extracting 3-D spatio-temporal features based on kinematic structure and surface representation of human body, and fusing multiple features at the kernel level using MKL technique. In addition, we present the experimental results on publicly available data sets. Finally Chapter seven presents a conclusion of our work and directions for possible future works.

Chapter 2

Related Work

In this chapter, we present an overview of previous related works for human action recognition. We start by reviewing the literature of 2-D based methods (i.e. sequence of RGB images). Afterwards, we review related works for 3-D based methods which are based on a sequence of depth maps for human action recognition. For further detail, we refer our readers to the following thesis's and surveys Klaser [13], Niu [14], Weinland et al. [15], Poppe [9], and Moeslund et al. [8] which are relevant to our work on generic action and activity recognition as well as motion analysis:

2.1 2-D based methods

According to the surveys by Poppe [9] and Moeslund et al. [8], most of the existing methods can be broadly classified into three categories:

- *Human model based methods* which employ 3-D (or 2-D) human model to recognize the action through the pose and the movement of human body parts.

- *Holistic methods* which use the knowledge about the localization and the structure of silhouettes in video to recognize the human action.

- *Local feature methods* which are entirely based on multiple local regions in each frame. These local regions are used to extract local features to recognize human actions. These features are combined together for the whole video to create an action descriptor.

2.1.1 Human model based methods

Human model based methods recognize actions by employing either human limbs movements, trajectories of joint positions, or landmark points on the human body parts. Recently, due to the advances in computer vision algorithms, marker-less motion capture systems using a monocular view have attracted the attention of many research groups [16, 1, 2, 17, 18, 19]. For instance, Ali et al. [16] applied skeletonization to silhouettes to obtain 2-D stick models, and their main joint positions are connected to joint trajectories. A sequence is represented by a set of chaotic invariants of these trajectories, and classified based on exemplars with a kNN-classifier. In the method proposed by Althloothi et al. [19], skeleton model, which employing information of human body parts as well as movements, were fitted into silhouettes to capture the positions of distal limb segments (i.e. arms, legs and head). The positions of distal limb segments were used as features. Then, Gaussian mixture models were used to model the spatio-temporal distribution of the distal limb segments over the period of an action.

Since the detection of body parts is a difficult problem in itself, other approaches are used to model the action based on set of points along the contour of the silhouette. For instance, Chen et al. [1] and Fujiyoshi et al. [20] developed silhouette-based approaches. They proposed a star skeleton representation based on the shape geometry of the silhouette (contour) to recognize human action. They detected the extremities of the contour with respect to the centroid and assumed that these points represent the head, hands, and feet. The center of mass of a human silhouette and extreme points corresponding to extreme contour points are detected as local peaks to represent the body as a single star.

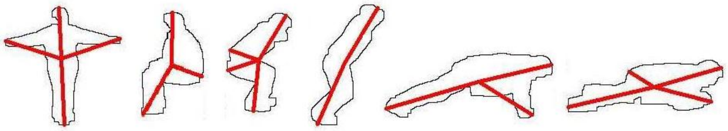


Figure 2.1. The star skeleton result (Chen et al. [1]).

In order to accurately detect extreme points for human body, Yu and Aggarwal [2] proposed a two-star skeleton representation by adding the highest contour point as the second star, as shown in Figure 2.2. Two sets of local peaks are estimated to find more precise extreme points. This work was modified later to variable star skeleton by Yu and Aggarwal [17]. The variable star skeleton was proposed to improve the accuracy of finding extreme points. They first constructed a medial axis from the human body contour. Then, they treated all junction points of the medial axis as stars. For each star, they generated a set of extreme points; each extreme point was processed according to its robustness, visibility, and proximity to its neighbor.

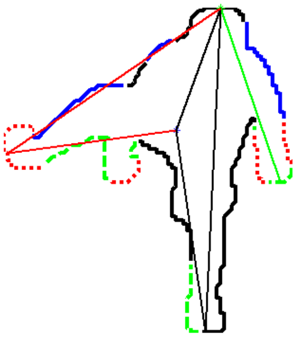


Figure 2.2. Two star skeleton (Yu and Aggarwal [2]).

Another example is the work of Chun et al. [18]. They proposed 3-D star spatio-temporal pattern based on the shape boundary information of the human posture using eight projection maps from different views. They detected the extremities of the silhouette with respect to the centroid as a shape features. They assumed that these extremities represent the head, hands, and feet. They accumulated the motion history of these features in order to create spatio-temporal pattern. Although a star shape is simple and fast for computation, its accuracy for detecting limbs needs further improvement. The results in these methods remain limited to constrained video data and can not be applied in a realistic video because the segmentation and localization of body parts in 2-D video still a difficult problem in itself.

2.1.2 Holistic methods

Holistic methods proceed in a top-down fashion: First, the whole body of person is localized in the ROI which is usually obtained through tracking or background subtraction. Then, the ROI is encoded in order to create an action descriptor that can describe human body movements. Consequently, the action descriptor is learnt via a global body movements without any notion of body parts. In other words, Holistic methods use global features such as optical flow, silhouettes structure or edge maps to represent the action descriptor trough number of frames in video. All these features (global features) are sensitive to partial occlusions and view angle variations because global features constrain on the structure of the whole body. To overcome these issues, multiple images from different view angles can be utilized to create a complete structure of human body (3-D data), or by using grid-based approaches which can be used to divide the observation into cells spatially [9].

In general, Holistic methods are based on global features for the whole body structure and do not require the localization of body parts. Most of existing automatic Holistic methods which are based on sequence of 2-D images can be categorized into two main approaches; appearance-based approaches and silhouette-based approaches.

Appearance-based approaches [5, 21] utilize the intensity of pixels or color configuration within the whole body to recognize human action. These approaches are significantly affected by the variances of clothing and body postures. In the silhouette-based approaches [3, 22, 23, 24, 25, 4], the whole body is located using the internal points of the silhouette or external points that can be detected along the contour of the silhouette. The key idea is that, the global dynamics of a silhouette via number of frames are utilized to recognize the human action.

In the past decades, the silhouette-based approaches using 2-D videos, which are captured by regular RGB camera, are widely used in human action recognition. In this context, there are many different approaches that are based on statistical shape analysis of silhouette as a function of time. The shape analysis aims to describe and locate the changes in human body shape. For instance, Blank et al. [3] analyzed space-time volume of silhouettes to create shape features for human action recognition, as illustrated in Figure 2.3. Cohen and Li [22] presented a 3-D spatio-temporal features for classifying and identifying human posture using SVM. They proposed a global features that are invariant to rotation and translation. The main advantage of their approach is in its ability to capture human shape variations allowing for the identification of body postures, but it needs to use multiple cameras to create 3-D spatio-temporal volume. Another example is the work of Yilmaz and Shah [23]. They track the contour points of a human over time to construct spatial-temporal volume, which can be treated as 3-D object in a space-time volume. Then, they analyze spatial-temporal volume by using different geometrical surface properties to describe the action.

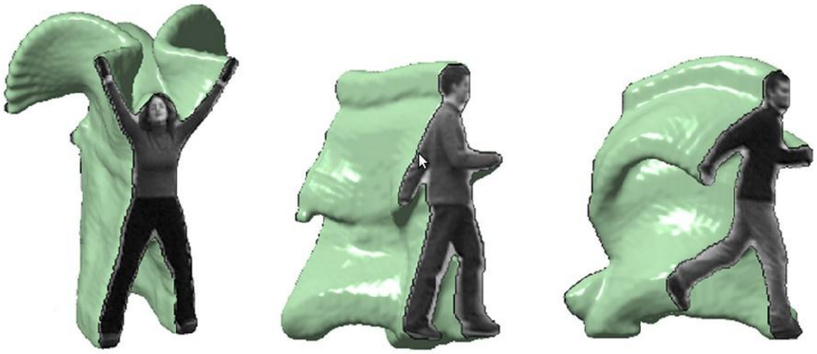


Figure 2.3. Space-time volumes for action recognition based on silhouette information (Blank et al. [3])

Recently, Venkatesha and Turk [24] defined a shape descriptor for interest points on the detected contour points and built an action descriptor using a Bag of Features (BoF) method. They used the temporal relation among matching interest points across successive video frames to train/classify the action in the scene. The limitation of this method is the difficultly to implement one-to-one matching among corresponding points on the contour.

Yilmaz and Shah [25] represented the action as a spatio temporal volume defined on structure of contour across successive video frames. The action volume described by analyzing the differential geometry of the surface volume. The limitation is an exhaustive search required for finding the correspondences between two volumes to classify the action.

Yan et al. [4] presented an approach called Action Feature Model (4D-AFM) from arbitrary views using four dimension (x, y, z, t) as shown in Figure 2.4. The modeling process starts with reconstructing 3-D surface volume. Then, spatio-temporal action features are computed in each view by analyzing the differential geometric surface properties of spatio-temporal volumes. Actions are recognized based on the scores of matching action features. Although 4D-AFM approach achieved high recognition rate, it had many limitations such as multi-cameras, synchronization and high computational complexity.

Bobick and Davis [5] use the idea of temporal templates for action recognition. They used both the motion energy images (MEI) which are binary masks that indicate regions of motion, and motion history images (MHI) which weight these regions according to the time when they occurred (higher weight for more recent motion), as illustrated in Figure 2.5, to represent the action.

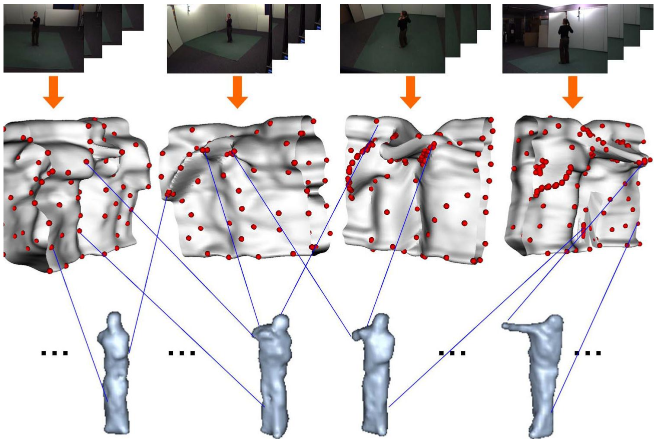


Figure 2.4. Illustration of constructing 4D-AFM. The second row shows the locations of the spatio-temporal action features on the surface of STVs. These action features are mapped to the 4D action shape as shown in the third row (Yang et al. [4])

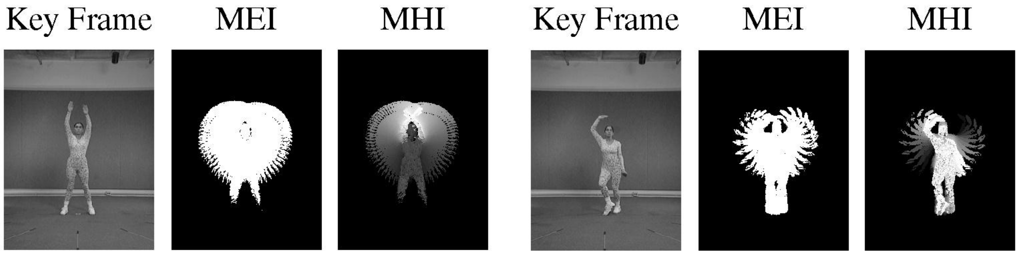


Figure 2.5. Bobick and Davis method for computing motion history images (MHI) and motion energy images (MEI) (Bobick and Davis [5]).

2.1.3 Local feature methods

Local feature methods are used to describe an actor as a collection of independent local descriptors. Local feature methods are obtained in a bottom-up fashion: First, spatial-temporal interest points are detected. Then, local patches or regions are calculated around these interest points. Afterwards, the local patches are combined into a final image representation. Local feature methods are less sensitive to noise and partial occlusion compared with Holistic methods because Local feature methods focus only on relevant interest points and the correlation among them.

In contrast to the Holistic methods, the local feature methods [26, 27, 28, 29, 30, 6] capture characteristic shape and motion information for a local region in video while Holistic methods capture the global body movements for a whole region. Also, local feature methods describe the actor as a collection of patches. These patches which correspond to interesting motions are sampled at space-time interest points where the local neighbourhood has a significant variation in both spatial and temporal domain. The idea of local feature methods is to collect the interest points as spatio-temporal features which are distinctive and descriptive enough the classify human actions. Therefore, the interest points detection approaches play an important role in local feature methods in order to classify the action. A key advantage of local features based methods is their flexibility with respect to the type of video data and does not require labeling or detecting body parts. Also, it is less sensitive to viewpoint, noise, and occlusions [8].

An example of local feature methods is the work done by Laptev and Lindeberg [26]. They extended 2-D Harris corner detector to 3-D as (x, y, t) and utilized it to detect points with high intensity variations based on velocity changes. This detector was capable to produce distinctive interest points to recognize human actions. Another example can be found in Efros et al. [27]. In their work, they build their motion descriptors using optical flow that relies on spatio-temporal correlations between two consecutive frames. Then, they matched the motion descriptors to a database of images to report the result based on the distance.

In the related work by Scovanner et al. [28], the SIFT descriptor [31] was extended to 3-D SIFT. The 3-D SIFT descriptor, which accurately captures the spatial-temporal features of the video data, is a spatio-temporal histogram-based representation of image patches. The 3-D SIFT descriptor used to recognize human action with a bag of words paradigm. Wu et al. [29] proposed an image representation called SIFT Motion Estimation (SIFT-ME). SIFT-ME is derived from SIFT correspondences in a sequence of video frames and adds tracking information to describe human body motion in both spatial and temporal domain. They utilized SIFT parameters for translation and rotation to describe 2-D human body motion with SIFT-ME as spatial-temporal interest points. Another related work of SIFT-based approach to recognize human action is known as MoSIFT by Chen et al. [30]. In this work, they proposed MoSIFT algorithm to detect and describe spatial-temporal interest points. The MoSIFT algorithm detects spatially distinctive interest points through local appearance to describe human body motion in both spatial and temporal domain.

Tian et al. [6] employed Harris detector and local HOG descriptor on Motion History Images (MHI) to perform action recognition and human detection. First, they detect interest points as the two-dimensional Harris corners with recent motion detected by MHI. Then, a global and local spatial motion smoothing filter is applied to the gradients of the MHI to eliminate noisy motions. They characterize the spatial and temporal features by histograms of oriented gradient in the intensity image and the MHI, respectively, and use a Gaussian-mixture-model-based classifier for action recognition, as illustrated in Figure 2.6.

2.2 3-D based methods

Compared with the existing aforementioned 2-D based methods for human action recognition, 3-D based methods rely on features extracted from RGBD sensor which includes more information on human body shape than a regular RGB cam-

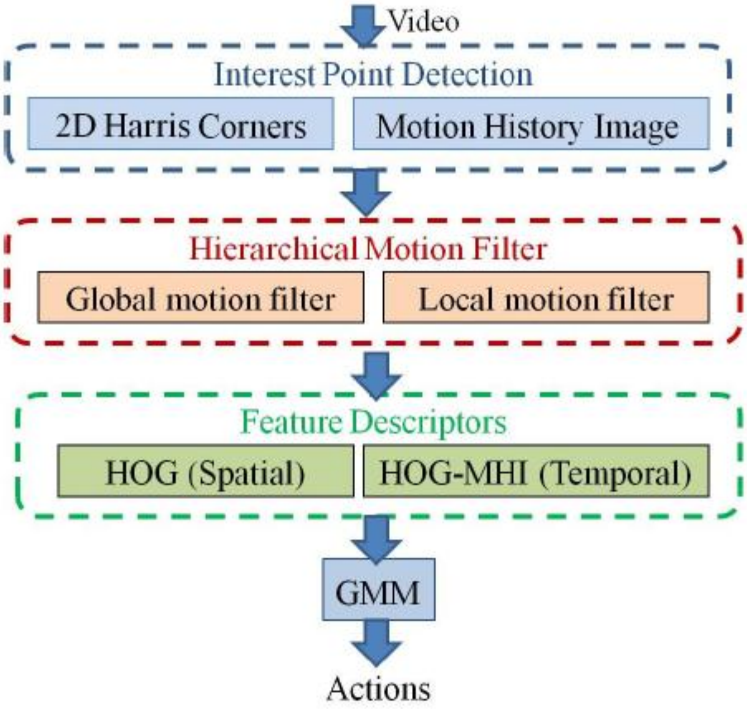


Figure 2.6. The framework of Tian et al. [6] method for action recognition in crowded videos.

era. Early, most of the 3-D based methods were used the range data as source of 3-D data. These 3-D based methods were build on a range data using laser range sensors which is expensive and slow at acquiring data to obtain a 3-D representation.

Recently, with the release of Microsoft Kinect SDK, research of human activity recognition based on sequence of depth maps has been explored. The Kinect provides both RGB images and depth maps which are robust to the change in illumination and color. In the last two years, There are a few numbers of researches utilized Kinect sensor in human limbs detection [32], and activity recognition from depth maps [12, 33, 7, 34]. For instance, Shotton et al. [32] propose a method to quickly and accurately predict 3-D positions of body joints from a single depth image, using an object recognition approach. In human action recognition, Wang et al [12] proposed a model for human actions, called the Actionlet Ensemble Model which is learnt using MKL technique to represent each action and to capture the intra-class variance. Based on the depth data and the estimated 3-D joint positions, they propose a feature called Local Occupancy Pattern (LOP) which can be treated as depth appearance for each 3-D joint. They utilized also the data mining solution to discover discriminative joints to represent Actionlet Ensemble Model, which is a linear combination of the actionlets, and their discriminative weights are learnt via a MKL technique.

Xia et al [33] developed a method, which is based on 3-D joint positions estimated from depth map, to create a histograms of 3-D joint locations using a spherical coordinate system. Then, they modeled the temporal evolutions of 3-D joint positions by discrete hidden Markov models in order to train/classify the action. Li et al. [7] proposed a Bag of 3-D-Points model for human action recognition. In order to select the representative 3-D points from depth map, they first projected depth maps onto three orthogonal Cartesian planes (XY, XZ, and YZ planes). Then, they sampled 2-D points at equal distance along the contours of the three projections, as illustrated in Figure 2.7. The sampled 2-D points were used to characterize the posture in each frame in order to classify the activity.

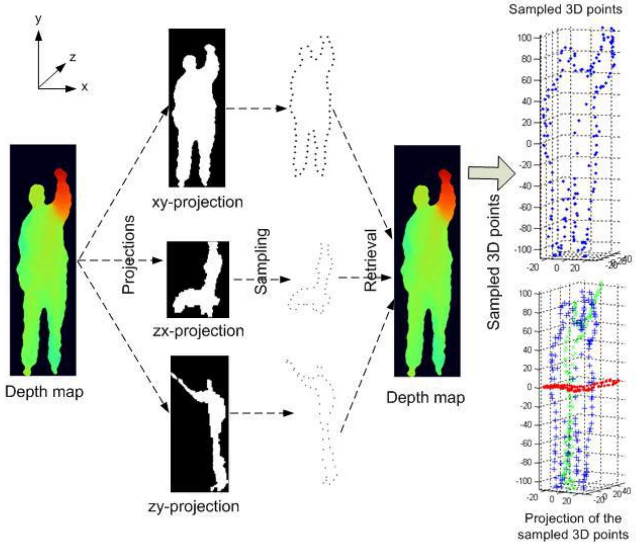


Figure 2.7. The process of sampling 3-D points from a depth map (Li et al. [7]).

Similar approach, Yang et al. [34] generated the Depth Motion Maps (DMM) from three orthogonal planes and accumulate global activities through entire video sequences. Then, Histograms of Oriented Gradients (HOG) are computed from DMM as the representation of an action model. Sung et al. [35] compute a set of features based on human pose and motion from the 3-D joint positions provided by Prime Sense with the Kinect. They proposed a hierarchical maximum entropy Markov model (MEMM), which considers a person's activity as composed of a set of sub-activities, with two-layered graph structure to train/classify different actions.

Yang et al. [36] proposed a method based on position differences of joints. They applied Principal Component Analysis (PCA) to differences of 3-D joint positions to obtain EigenJoints by reducing redundancy and noise. Then, they employed the

Nave Bayes Nearest Neighbor (NBNN) classifier for multi-class action classification. Zhang et al. [37] proposed a 4-D local spatial-temporal descriptor that combines both intensity and depth information. The feature descriptor concatenates the intensity and depth gradients within a 4-D hyper cuboid, which is centered at the detected feature point, as a feature. Zhao et al. [38] extract the feature vector of each video clip by combining RGB-based descriptors and depth-map based descriptors. They used three types of features: Local Depth Pattern (LDP) which describes the local region of interest points in depth map, HOG and HOF features in RGB data. All these features are combined in one vector in order to classify human action as multi-class classification using LibSVM [39].

2.3 Summary

From the literature review, we have observed that the depth map contains rich shape information of the human body compared with RGB image. The depth map is invariant to colour, texture and illumination variation and can capture the changes in the body pose. Also, we have observed most of the shape features are carried either by the interesting points on the contour or surface points on the silhouette. The surface points which have more information than the contour, can describe the structure of the silhouette. These observations form the core concept of the method presented in chapter five to represent the body shape using surface points provided by the depth map. Our proposed method is extended to a spherical harmonics representation in a novel way to describe the shape/pose of the human body.

In addition, we have observed that the distal limb segments, such as forearms and shins, provide sufficient and compact features for human action recognition. The movement of distal limb segments provides the key to estimate 3-D motion features of human body. Also, we have observed most of the methods in the literature are based either on structure of silhouette or 3D joint positions, and there are redundancy in using the joints of the human body. These observations form the core concept of the method presented in Chapter Four. Therefore, the main concept of our work in this dissertation is to incorporate the surface representation based method and kinematic structure based method in a novel way to enhance the classification accuracy.

Chapter 3

Fusion Strategies

In fusion strategies, incorporating information is accomplished by utilizing the information available from multiple modalities, multiple features, or multiple sources in order to increase the accuracy and robustness of classifiers. Multiple information can be incorporated at different levels, including the sensor level, feature level, score level or decision level. In this chapter, we focus on the fusion strategies at the feature level because it is relatively a new topic while fusion at the match score and decision levels have been extensively studied in the past decades. Consequently, this chapter presents a general overview of fusion methods at different levels, then a feature fusion level for the task of pattern classification is presented in detail.

3.1 Introduction

The classical definition of the word ”fusion” refers to incorporating information available from different modalities. Figure 3.1 shows various types of fusion methods that can be used in the human action recognition. In general, fusion can be categorized as: 1) fusion before matching and 2) fusion after matching [40]. The fusion before matching integrates information either at the sensor level or at the feature level. In the sensor level fusion, the raw data from the sensors are combined while in the feature level fusion, different extracted sets of features from multiple modalities are combined together. On the other hand, fusion after matching integrates information either at the match score level or decision level by employing dynamic classifier selection or voting scheme.

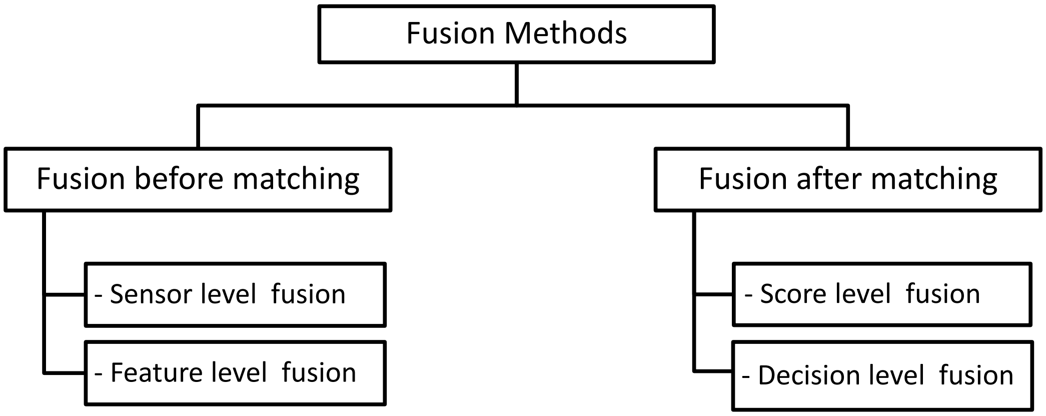


Figure 3.1. Fusion can be accomplished at various levels.

In summary, fusion can be categorized in to four levels:

1. *Sensor level fusion*: Multiple data acquired from different sensors can be processed and integrated to generate a new type of feature.

2. *Feature level fusion*: Multiple feature sets can be fused to create a new

descriptor. The new descriptor is calculated from two or more sets of the original feature sets for classification task.

3. *Score level fusion*: Multiple scores from different classifiers were matched and fused to generate a single scalar score.

4. *Decision level fusion*: Multiple class labels were fused by employing techniques like majority voting to select a single class label.

Although multiple features are noisy, overlapping, and confusing but fusing multi-features in a suitable way will produce a better classification performance than a single feature [41]. Hence feature fusion is an essential step for any successful operation in pattern classification. It is necessary to identify the useful features and remove undesired features which reduce the performance of the classifier before fusion operation in order to fuse only the rich features that have a valuable. Thus, features which are confused with the rest of the set and weakly correlated to the class labels should be removed through the fusion operation to avoid the confusion in the classifier. Therefore, selecting useful features is the key to any feature fusion operation, and play an important role for any successful pattern classification task. In this dissertation, we focus on the fusion at the feature level since features are the most important cues for any successful classification task.

3.2 Feature level fusion

In the last decade, a significant amount of research has been done on different fusion techniques but only a few of them reported for feature level fusion techniques [42, 43, 44]. Due to limited number of research on feature level fusion, it is still not clear how to select useful features to provide better classification results. Most of feature fusion operations deal with the fusion operation by removing redundant and irrelevant features. For instance, let's assume we are given a set of N training samples $\{(x_i, z_i, y_i)\}_{i=1}^{N}$, where x_i is a feature vector of the i^{th} sample in the training set $\mathcal{X}$ with dimension D_1, $x_i \in R^{D_1}$ while $z_i \in R^{D_2}$ is another feature vector of that sample in the set $\mathcal{Z}$, and $y_i \in \{-1, +1\}$ is their corresponding class label. Therefore, if two features $\{(x_i, z_i)\}_{i=1}^{N}$ have similar distribution, one of them is redundant. Also, if one of the feature correlates poorly with the class label $\{(y_i)\}_{i=1}^{N}$, the feature is considered to be irrelevant. The final set of features is fused with useful features to obtain a better classification accuracy.

Feature level fusion involves the integration of multiple types of features which are extracted from multiple modalities. Since the feature set contains rich information about the subject, fusion at this level is expected to provide better recognition rate than the match score level or decision level. However, feature level fusion is difficult to implement in practice because of the curse of dimensionality problem and incompatibility [40]. The curse of dimensionality problem will increase the computational time as result of features size. Therefore, feature fusion operation should consider those issues in the case of using multiple features instead of single feature. Consequently, the main aim for all feature fusion operations is to remove the overlapping features, undistorted features and to merge only the useful features that have a valuable in order to solve the curse of dimensionality problem.

Solving the curse of dimensionality problem is one of the strongest motivation of the feature fusion operations. In other words, fusion different types of features is a process to generate a new descriptor by combining useful features and removing useless features. Hence, the main core of feature fusion operation lies in combing the useful features in order to avoid the curse of dimensionality problem and to enhance the classification performance. In the next section, we present a novel technique which optimally combines the features using MKL technique by learning multiple kernels instead of using single kernel.

3.2.1 Feature fusion using MKL-based SVM

MKL-based SVM has recently received great attention in the field of machine learning, and provides a general framework for learning multiple features using multiple kernel functions. MKL-based SVM aims to optimally combines and utilizes multiple kernel functions and features instead of using a single kernel function in learning kernels based SVM classifier.

SVMs [45] are the widely used classifiers for human action recognition [46, 47, 48]. SVM classifier utilizes a hyperplane that discriminate two classes of data in feature space, where the margin between the two classes is maximized between the nearest data point of each class as shown in Figure 3.2. This linear classifier is termed the optimal separating hyperplane because it maximises the margin distance.

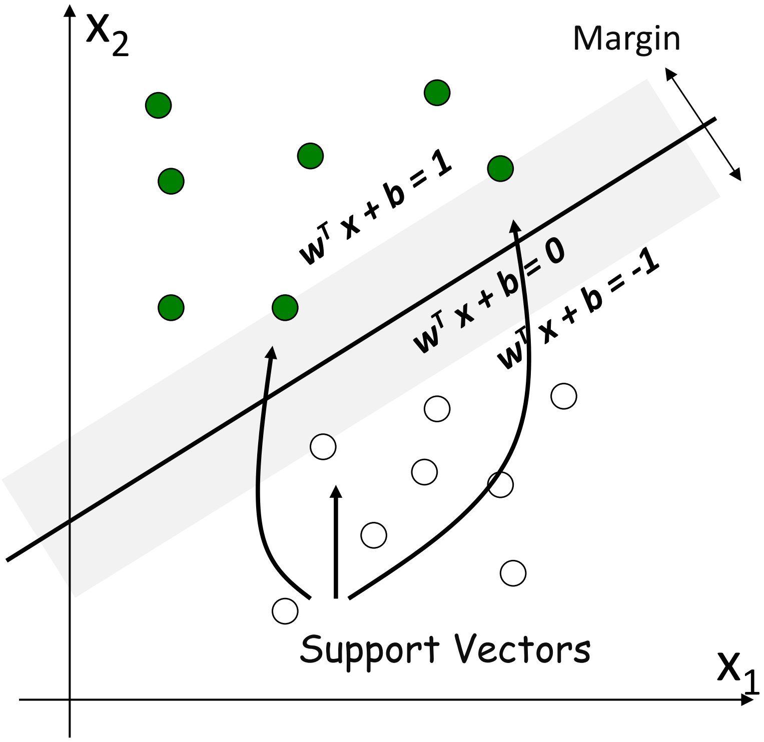


Figure 3.2. The SVM classifier utilizes a hyperplane to discriminate two classes of data in feature space.

Whereas the original problem of SVM classifier is stated in a finite dimensional space, where the discriminate sets are not linearly separable in that space. For this reason, it was proposed that the original finite-dimensional space be mapped into a much higher-dimensional space, presumably making the separation easier in that space as illustrated in Figure 3.3. This mapping is designed to ensure that dot products are computed in terms of the variables in the original space, by defining them in terms of a kernel function. Thus, Kernel functions are used to accommodate non-linear functions and project feature vectors onto more discriminative spaces. Also keep in mind that solving the SVM problem involved computing the dot products between all pairs of training points using kernel function.

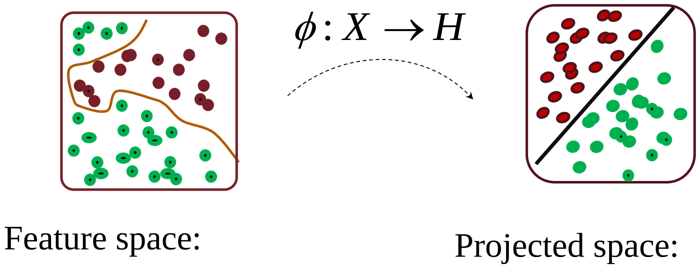


Figure 3.3. The original finite-dimensional space be mapped into a much higher dimensional space making the separation easier.

3.2.2 Literature review of MKL

Recent researches have shown that using MKL instead of a single kernel function can enhance the performance. Senechal et al. [49] show that using multiple kernels and different types of features instead of a single kernel with a single type of feature can enhance the discrimination power of the classifiers and improve the performance of SVM. The works presented in [48, 50] show that MKL can improve the performance of classifiers during human action recognition. The idea behind

MKL is to optimally combine different kernel matrices calculated based on multiple features with multiple kernel functions. The authors of [51] justified the superiority of MKL-based SVM over canonical SVM by showing that the learned discriminant hyperplane of MKL-based SVM performs better than canonical SVM. Within this framework, the problem of feature data representation through the kernel in SVM is transferred to set the optimal value of kernel combination weights. Therefore, the MKL based SVM is defined to learn both the decision boundaries between data (features) from different classes and the kernel combination weights in a single optimization problem.

Several MKL algorithms, as surveyed in [52], have been applied to affective analysis and achieved better recognition results compared to single kernel SVM with a single type of feature. There are several methods for solving the optimization problems in MKL-based classifiers. Methods such as [53, 54], called ”Two-Step methods”, use two nested iterative loops to optimize both the classifier and kernel weights. Usually, in the inner iteration a solver of kernel-based classifiers such as SVM is implemented by fixing the parameters of kernel combination function while in the outer iteration the function's parameters are learned with fixed parameters of the classifier. In our method of solving the MKL based multiclass-SVM, the Two-Step method described in [52] is used, during which a C-SVM solver with one-against-one rule is applied in the inner loop to find the decision boundary between each pair of classes while the combination weights of multiple kernels with different kernel parameters are updated in the outer loop by the SimpleMKL algorithm [54].

Recently, MKL technique as surveyed in [51] have been proposed for feature fusion at the kernel level within kernel-based classifiers. MKL is used to optimize kernel weights while training the SVM. Therefore, we applied MKL technique to combine different types of features using SimpleMKL algorithm. In SimpleMKL algorithm, an alternative approach for fusing features is to combine kernel functions to create a combined kernel function which is defined as the weighted sum of the multiple kernels.

3.2.3 Advantage of using MKL

There are several reasons for preferring a multiple kernels over a single kernel. It is mainly done to improve the accuracy and robustness of the classification system. In some cases such as the classification among multiple classes, by using one kernel function, the projected features may not be distinguishable for all classes whereas using another kernel function may produce better results in several classes. Furthermore, different features will have different distributions, and it is hard to represent a combination of them using a single kernel. Therefore, a single kernel cannot perform well when the nature of features are incompatible.

In summary, the most important advantages of using MKL are as follow: (1) A multiple kernels will be an efficient approach for multiple types of features that can be trained with multiple kernels and the outputs are combined in order to achieve high performance. (2) Another reason for using multiple kernels is to improve the generalization performance: one type of kernel may not perform well for a certain input when it is trained with multiple types of features. Finding a single kernel to work well for all features is difficult. Hence, multiple kernels can be used to combine different types of features and to give a better performance than a single kernel. (3) Solving the curse of dimensionality problem is one of the most important advantage of using MKL technique. Based on the weights, MKL fusing the useful features in a combined kernel matrix which is defined as the weighted sum of the individual kernels. The main core of MKL lies in optimizing kernel weights which are used to avoid the curse of dimensionality problem while training the SVM.

3.3 A framework of MKL-based multiclass-SVM

In this section, we focus on formulating methods for solving the optimization problem based on MKL-based multiclass-SVM [54].

3.3.1 MKL-based SVM

MKL-based SVM algorithm uses a combination of different kernels with different parameters correspond to different representation of multiple features, and defines an automatic learning method that picks the optimal parameters. MKL-based

SVM was proposed to solve the primal form of the optimization problem defined in Equation 3.1.

$$
\begin{aligned}
\min_{w,\xi,d} \quad & J(w,\xi,d) = \frac{1}{2}\sum_{m=1}^{M} \frac{1}{d_m}\|w_m\|^2 + C\sum_{i=1}^{N}\xi_i \\
s.t. \quad & y_i(\sum_{m=1}^{M} w_m^T \phi_m(x_i) + b) \geq 1 - \xi_i, \quad i = 1,2,...,N \\
& \xi_i \geq 0, \quad i = 1,2,...,N \\
& \sum_{m=1}^{M} d_m = 1, \quad d_m \geq 0, \quad m = 1,2,...,M
\end{aligned}
\tag{3.1}
$$

where w_m is the direction of the hyperplane in each projected mapped , w denotes the set $\{w_m\}$ and M is the number of projected maps in use. In addition, $d = (d_1, d_2, ..., d_M)^T$ controls the weight of the squared form of w_m in the objective function.

The optimization problem defined in Equation 3.1 is solved based on its associated dual form shown in Equation 3.2.

$$
\begin{aligned}
\min_{d}\max_{\beta} \quad & L(d,\beta) = -\frac{1}{2}\beta^T(\sum_{m=1}^{M} d_m K^{(m)})\beta + y^T\beta \\
s.t. \quad & \sum_{i=1}^{N}\beta_i = 0, \quad 0 \leq \beta_i y_i \leq C, \quad i = 1,2,...,N \\
& \sum_{i=1}^{M} d_m = 1, \quad d_m \geq 0, \quad m = 1,2,...,M
\end{aligned}
\tag{3.2}
$$

where $y = (y_1, y_2, ..., y_N)^T$, $\beta_i = \alpha_i y_i$, and $K_{i,j}^{(m)} = k_m(x_i, x_j) = \langle \phi_m(x_i), \phi_m(x_j) \rangle$ as denoted in Equation 5.11. Therefore, the kernel matrix $K^{(m)}$ calculated from each projected map, and combined linearly based on kernel combination weight vector d_m as $K = \sum_{m=1}^{M} d_m K^{(m)}$, where $d = (d_1, d_2, ..., d_M)^T$ is the kernel combination weight vector, M is the total number of kernel functions in use, and $\beta = (\beta_1, \beta_2, ..., \beta_N)^T$ is known as the vector of Lagrangian coefficients in SVM.

In this manner, Two-Step method [53] is used to solve the optimization problem defined in Equation 3.2, where two nested iterative loops are set to optimize both the classifier and kernel combination weights. In the inner iteration a solver of SVM is implemented by fixing the vector of kernel combination weights while in the outer iteration a reduced gradient descent algorithm [55] with the golden section search method [56] is used to update the combination weights with fixed parameters of the SVM classifier. The main core of MKL-based SVM lies in optimizing kernel weights while training the SVM classifier.

For instance, let's assume we are given a set of training samples $\{(x_i, z_i, y_i)\}_{i=1}^{N}$, where x_i is a feature vector of the i^{th} sample in the training set $\mathcal{X}$ with dimension D_1 while $z_i \in R^{D_2}$ is another feature vector of that sample in the set $\mathcal{Z}$, and $y_i \in \{-1, +1\}$ is their corresponding class label. $k(\cdot, \cdot)$ is a kernel function that maps two feature vectors to be a positive scalar. each feature should be linearly combined with at least two kernel functions from three kernel functions: Gaussian function, polynomial function and radial basis function (RBF), each with different parameters. To show how these features are linearly combined, suppose we are given a pair of samples (e.g., the i^{th} and j^{th} registered images), the fusion of $(\{x_i, x_j\})$ features and $(\{z_i, z_j\})$ features at kernel level within the MKL-based SVM framework is handled as follows,

$$K_{i,j} = \sum_{m=1}^{M} (d_m k_m(x_i, x_j) + d_{m+M} k_m(z_i, z_j)) \tag{3.3}$$

Hence in this fusion work, the dimensionality of kernel combination weight vector d is $2M$.

Therefore, once given a new test sample, its label (determined by y_0) can be calculated according to the follow function:

$$y_0 = \text{sgn}[\textstyle\sum_{i=1}^{N}\sum_{m=1}^{M}\beta_i(d_m k_m(x_i, x_0) + d_{m+M} k_m(z_i, z_0))] \tag{3.4}$$

3.3.2 MKL based multiclass-SVM

Two techniques are commonly used in the literature to classify U classes using binary classifiers: one-against-one and one-against-rest. In the one-against-one technique, $U(U-1)/2$ binary classifiers are built from all pairs of distinct classes, whereas in the one-against-rest technique U binary classifiers are built for each class of data. The authors of [54] presented a structure of MKL based multiclass-SVM algorithm in the outer iteration of the Two-Step method. In their structure, a single kernel combination weight vector is jointly learned for all binary classifiers in the multiclass-SVM, and the general objective function $L(d)$ is defined in Equation 3.5.

$$L(d) = \sum_{u \in \Phi} L_u(d) \tag{3.5}$$

where Φ is the set of all pairs of distinct classes considered in the multiclass-SVM, and $L_u(d)$ is the object function of a binary MKL-based SVM defined in Equation 3.2 with fixed β for each binary classifier.

By this definition, the inner loop of the MKL based multiclass-SVM is to solve the multiclass-SVM while in the outer loop a single kernel weight vector is learned to minimize the summation of the objective functions from all binary classifiers. Therefore, the learned optimal kernel weight vector can be used for all binary classifiers, which generally increases the recognition result of multiclass-SVM. In our work, the one-against-one technique is used based on the experiences of the work in [57], and the classification of novel samples is done by a max-wins voting strategy.

3.4 Summary

In this chapter, we have presented the concept of fusion strategies which are achieved by utilizing the information available from multiple modalities, multiple features, or multiple sources in order to increase the accuracy and robustness of the classifier. A general overview of fusion techniques in different levels is presented, then a feature fusion level using MKL technique for the task of pattern classification is presented in more detail.

Fusion operations can be adopted either before matching or after matching process. Fusion before matching integrates information either at the sensor level or at the feature level, while fusion after matching integrates information either at the match score level or decision level. Since the feature set contains rich information, different types of features as multiple features in any recognition task will enhance the classification performance. Thus, feature level fusion is expected to provide better recognition results than the other level especially if the features used for the classification are discriminating, undistorted and independent. Recently, MKL-based SVM is used to fuse multiple features at the kernel level using multiple kernel functions. The idea behind MKL technique is to optimally combine different kernel matrices calculated from multiple features with multiple kernels. The calculated kernel matrices are linearly combined based on corresponding kernel combination weights in our MKL framework. Therefore, the MKL-based SVM is defined to learn both the decision boundaries between data from different classes and the kernel combination weights in a single optimization problem.

Chapter 4

Kinematic Features for Human Action Recognition

This chapter presents a method for human action recognition using kinematic structure of human body. Since the motion of human limbs is an important source of information for classifying human actions, a set of kinematic features that are derived from the motion of human limbs are used to represent human actions in videos. Within this frame, kinematic features are adopted as motion features that can be used to describe complex actions. The idea behind using kinematic features is to convert human skeleton model (kinematic structure) into a discriminative features that can improve the motion-based action classification. One of the challenges of extracting the kinematic features is to accurately detect body limbs and then describing body motion using the detected limbs.

To the best of our knowledge, a good approach for detecting/representing human body limbs is a human skeletonization (kinematic structure) which consists of line segments linked by joints. In this case, the motion of segments or joints provide the key to estimate the motion and recognize the action. In our work, this concept is achieved in a novel way by detecting distal limb segments from silhouette to accurately fit a human skeleton model. With the human skeleton model, we generalize variations among different people performing the same action. The proposed method is used to detect distal limb segments which are used to describe the human motion in visual data.

4.1 Detecting distal limb segments

Accurate limb detection in visual data can be a challenging task because of the invisibility of human limbs or their occlusion by other body parts in visual data. Most of the researchers use either an automatic methods or non-automatic methods to detect the human limbs. Automatic methods use marker-less motion capture systems to detect the human limbs, while non-automatic methods could involve manual interventions to set the joints in the image or involve optical markers on the actor to detect the limbs. Although non-automatic methods are more accurate than automatic methods, they are usually expensive and not suitable for surveillance purposes or remote monitoring systems [58]. Due to the advances in computer vision algorithms, marker-less motion capture systems using a monocular view has attracted the attention of many research groups. They used different approaches for marker-less motion capture system to detect the human limbs. Most often detected limbs are used to fit a human skeleton model to represent the body posture. This process is known as skeletonization. The process of fitting a skeleton model can be performed automatically or manually.

In our work, we present an approach for detecting distal limb segments for accurate fitting a human skeleton model in visual data for human action representation and recognition. Our approach for detecting the distal limb segment consists of the following steps: First, an adaptive multi-dimensional Gaussian model is utilized to segment a moving actor from static background. A simple tracker is applied to the moving actor (i.e., foreground) to detect the motion direction. The foreground which represents the body silhouette is considered as a blob. Then, a distance transform algorithm followed by a line fitting algorithm is used to fit a nine-segment skeleton model in the body silhouette. The fitting process is performed independently for each of the four limbs to avoid the combinatorial complexity problems. After the human skeleton fitting step, a six-element descriptor vector is extracted as kinematic features to characterize the statistical behaviour using GMM for action recognition. Figure 4.1 illustrates a general overview of our framework.

In summary, the contributions of our work are as follow: a novel method for fitting distal limb segments into nine-segment skeleton model, accurate limb extraction based on a nine-segment skeleton model, reduction of the skeleton to the distal limbs only for action recognition, evaluation of the accuracy of the approach with respect to limb detection and human action classification.

4.2 Skeletonization

Using the silhouette points to represent a human posture is inefficient since all the points in the silhouette are similar to each other and contour points provide only the structure of the silhouette. On the other hand, simple information from the silhouette like center, height and width of blob cannot represent a human posture and it is difficult to discriminate human postures. Consequently, human skeleton model seems to be a good way to represent a human posture. Also, it can be utilized to extract human postures variations to characterize human actions. In this section, we describe our approach for creating a human skeleton model from

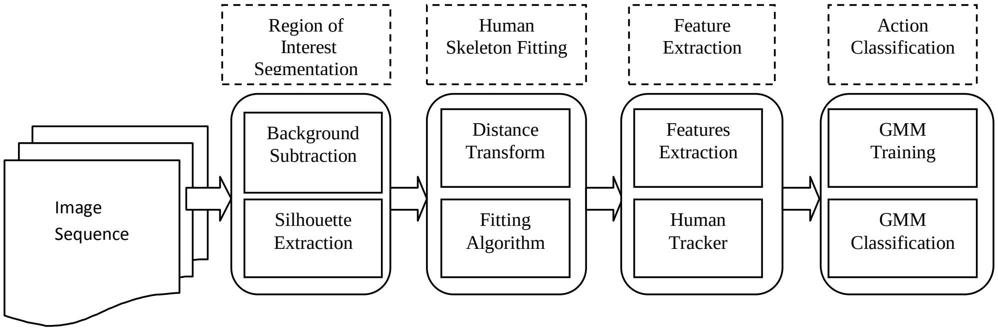


Figure 4.1. The process of extracting distal limb segments for human skeletonization in order to classify the action.

sequence of 2-D images followed by human action recognition based on kinematic features which are extracted from the human skeleton model.

In the following, we describe our approach for background segmentation, fitting a nine-segment skeleton model into the limb segments and kinematic features extraction. Afterwards, we present pose estimation of the human limbs and action classification using kinematic features and GMM.

4.2.1 Segmentation

Most vision-based human motion recognition systems start with Region of Interest (ROI) segmentation in temporal or spatial domain. ROI segmentation in video data aims at detecting regions that correspond to moving actors in a sequence of frames. For scenes with a relatively static background, most ROI segmentation methods are based on background subtraction algorithm, which calculates the difference between the current image and a reference background image, pixel-by-pixel, to detect moving actors in a sequence of frames.

In this dissertation, an adaptive Gaussian model using probability density functions for image pixels were built to evaluate if the pixels belong to background or foreground [59]. This approach is mainly utilized to segment moving actors from the static background to represent the ROI. The first few frames ($20 \sim 30$ frames) of each video are usually used to learn the covariance matrix and the mean value of each pixel. These values are used in the probability model to extract the ROI from the scene. This probabilistic model (the mean and covariance) is updated continuously for every new frame to adapt to changes occurring in the scene during an action performance. To increase the accuracy of the adaptive Gaussian model for background subtraction, we modified the algorithm of Stauffer and Grimson [59] by representing the color value of each pixel in the RGB image $I(i, j)$ using a multi-dimensional Gaussian distribution instead of a one dimensional Gaussian distribution for gray scale images. Thus, the probability $P(x_{t,i,j})$ of a background pixel (i, j) with color value $x_{t,i,j}$ at time t is defined as:

$$P(x_{t,i,j}) = \frac{1}{(2\pi)^{d/2}|\sum_{i,j}|^{1/2}}\, e^{-\frac{1}{2}(x_{t,i,j}-\bar{x}_{i,j})^T {\sum_{i,j}}^{-1}(x_{t,i,j}-\bar{x}_{i,j})} \tag{4.1}$$

where $\bar{x}_{i,j}$ is the mean and $\sum_{i,j}$ is the covariance matrix of pixel (i, j). Using this probability estimate, we compute the probability $P(x_{t,i,j})$ for pixel $I(i, j)$ in frame t. Then, we apply the following thresholding to obtain the foreground:

$$I(i,j) = \begin{cases} 1 & P(x_{t,i,j}) \leq T \\ 0 & P(x_{t,i,j}) > T \end{cases} \tag{4.2}$$

where T is an experimentally selected threshold. In our experiments, we assumed that the first few frames ($20 \sim 30$ frames) of the video contain only the background scene which used to select the threshold, T that works with the video.

4.2.2 Human skeleton fitting

For accurate skeletonization, we need to detect human body limbs with high accuracy at each frame because they play an important role in presenting the human behaviour. Once a binary image, which represents the silhouette (blob) of the moving body, is successfully extracted from the background, Medial Axis Transform (MAT) is used to convert the silhouette into skeleton structure as shown in Figure 4.2. The MAT is a process for reducing foreground regions in a binary image into a skeletal remnant that largely preserves the extent and connectivity of the original region while throwing away most of the original foreground pixels to produce the skeleton.

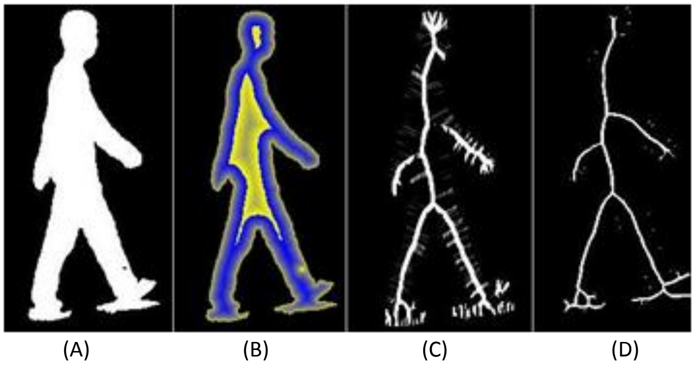


Figure 4.2. Illustrates the process of extracting skeleton; (A) Binary image, (B) Distance transform algorithm (C) Result of MAT algorithm, (D) Skeleton after applying morphological operations.

The MAT can be performed in two different ways as follows:

- A morphological thinning operator successively erodes away pixels from the boundary while preserving the end points of line segments until no more thinning is possible [60]. This is a repetitive and time consuming process of testing and deletion of each pixel. The result depends upon the shape of silhouette, the number of morphological operations, and the different orientations of the silhouette.

- The second method, which is used in this dissertation, is based on Distance Transform (DT) of each shape point to the boundary and tries to extract the skeleton by finding the points in the medial axis of the object. The result of the DT algorithm is a gray level image that look similar to the input image, except that the gray level intensities of points inside foreground regions are changed to describe the shortest distance from the edges of an object to a given shape point in the object [61]. DT is the radius of the largest disk in the shape centered in $p(x, y)$.

Mathematically, the medial axis, or skeleton $S(\emptyset)$ of a binary image, is a scalar field that contains the points that have maximal equal distance between two pixels $q(x, y)$, $r(x, y)$ on the boundary ∂ of the object $\emptyset$ and pixel $p(x, y)$ in the center axis with highest distance transform value:

$$S(\emptyset) = \{p \in \emptyset | \exists q, r \in \partial, q \neq r : dist(p, q) = dist(p, r)\} \tag{4.3}$$

These pixels are considered as medial axis of the object because they are located equidistant with at least two boundary points from both sides of the silhouette.

The MAT algorithm can be implemented with a high degree of parallelism for obtaining the skeleton of the image. MAT algorithm is faster than the thinning algorithm which iteratively removes the boundary layer from outside to inside. After obtaining the skeleton model using MAT, binary morphological operators, such as opening and closing operations are applied to the binary image which is extracted using MAT algorithm to filter out some of the disconnected pixels and also fill small gaps. Opening and closing operations are both derived from the fundamental operations of erosion and dilation. The basic effect of the opening operation is to remove some of the foreground pixels from the edges of regions of foreground pixels. The closing operation enlarges the boundaries of the foreground regions in the image and shrink the background holes in such regions. Hence, opening removes small islands and thin filaments of foreground pixels. Likewise, closing removes islands and thin filaments of background pixels. Overall, these two operations smooth the boundary of the binary image and fill small gaps. Figure 4.3 shows the results of background subtraction and skeleton extraction using MAT for sample images.

4.2.2.1 Fitting algorithm

Since MAT algorithm does not preserve the connectivity of the skeleton, line fitting algorithm is used to resolve this issue. Therefore, nine-segment skeleton model in each frame is created using its skeleton shape by utilizing the line fitting algorithm.

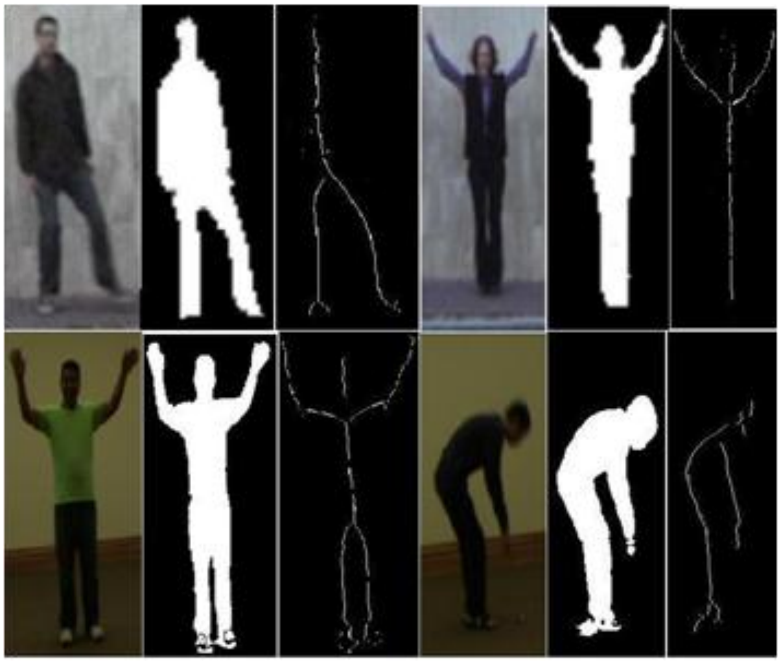


Figure 4.3. Illustrates the results of background subtraction and skeleton extraction using MAT algorithm.

The skeleton model used here to represent the kinematic structure consists of nine segments and six joints as shown in Figure 4.4. This human skeleton model is created by applying line fitting algorithm to the skeleton generated using the MAT algorithm.

In the line fitting algorithm, the line between the head center and the centroid (x_c, y_c) point of the whole body (hip center) is utilized to represent the torso of the body. The location of the head center is detected from the original image using a skin detection algorithm [62]. The algorithm initially converts the images from RGB color space to HSV color space. Then, the H channel is utilized to localize the range of colors found in human skin. By thresholding the value of channel H

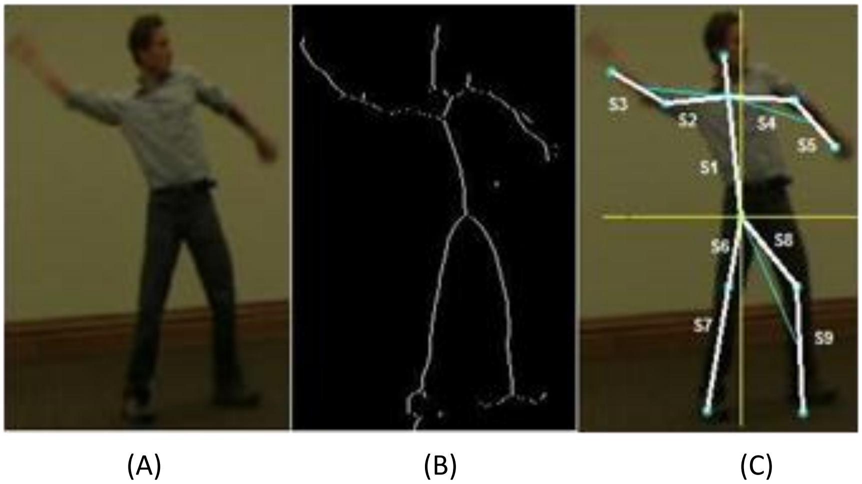


Figure 4.4. Human skeleton model for throwing action (A) original image (B) skeleton using the MAT algorithm (C) skeleton model using the fitting algorithm.

between 15 and 25, the head blob can be detected. Then, the center of the head, which is the center of the head blob, is used to represent the head position. If the head blob cannot be detected using the skin detection algorithm (e.g., in gray scale images), we then assume the head center is the pixel that had the same horizontal value x_c of the centroid (x_c, y_c) and the same vertical value of the highest white pixel in the range of ±10 pixels of the skeleton. This commonly occurs in the turning around action that the face is occluded and gray scale images where color skin detection algorithm cannot work. Examples are shown in Figure 4.5.

Since the skin detection algorithm depends on the channel H of the HSV color space, it cannot be applied to gray-scale images. There are other methods on finding human faces in images such as Haar features or templates matching [63] that can be utilized. In our work, we used skin detection algorithm as a fast and

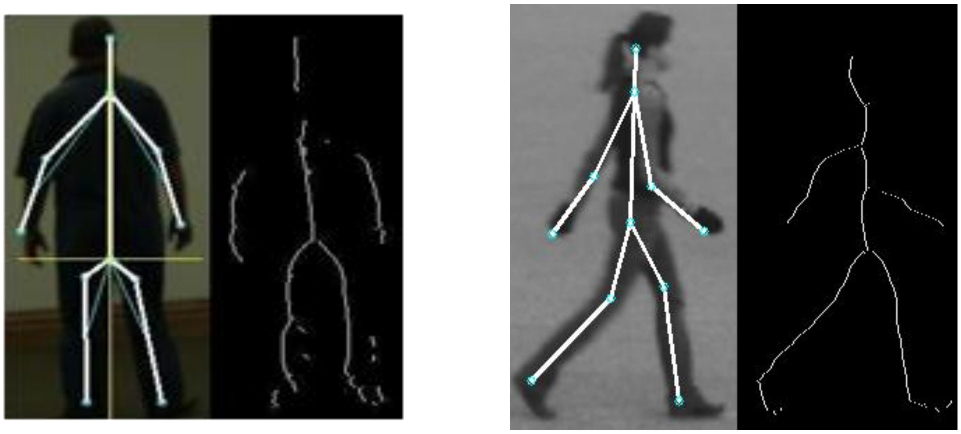


Figure 4.5. Skeleton extraction in the case if the head blob cannot be detected using skin detection algorithm.

efficient method to find potential facial skin regions in color images rather than applying more complex face detection techniques that increases the execution time of our action recognition method.

In the next step, which is a line fitting algorithm, the least squares curve fitting method is used to represent line with an equation rather than as separate points. To do this we use a process called line or data fitting.

In particular, for a line $y = ax + b$ and a set of N points (x_i, y_i), where $i = 1, 2, ..., N$ in 2-D plane, we aim to find the parameters a and b such that an error criterion that describes the deviation of the N points from the line is minimized. Thus, we utilized the vertical deviation e_i which is calculated as:

$$e_i = y_i - (ax_i + b) \tag{4.4}$$

Therefore, data with N points has a total error E:

$$E = e_1^2 + e_2^2 + e_3^2 + \ldots + e_n^2 = \sum_{i=1}^{N} e_i^2 \tag{4.5}$$

Substituting the equation of e_i yields,

$$E = \sum_{i=1}^{N} (y_i - (ax_i + b))^2 \tag{4.6}$$

In order to find the best fit, E is minimized using the standard least squares method, which attempts to minimize the sum of the squares of the error distances (E) for all the points. In the case of horizontal and vertical lines, the algorithm takes the average points of y_i and x_i to estimate the line. In the first step of the fitting algorithm, the line from the centroid (x_c, y_c) of the silhouette to the head center is fitted to represent the torso $S = \{S_1\}$. Once the torso has been successfully fitted, all four limbs are scaled based on the white pixels in the skeleton which is extracted from the silhouette. Each limb S_i is fitted independently to avoid the combinatorial complexity problems. Therefore, the bounding box of the silhouette is divided into four boxes based on the centroid (x_c, y_c) point to make sure that all the limbs are covered using the fitting algorithm.

After fitting an appropriate lines, which are based on the number of the white pixels in each box, all the lines are connected to the shoulder joint which is located at $1/4$ of the total distance from the head center to the centroid. These lines are used to identify the position of the legs and arms based on the location of the box. In the case where the torso angle is greater than ±30 degree, as in jumping, picking up and bending actions, the bounding box of the silhouette is divided into two boxes based on the centroid (x_c, y_c) point of the silhouette. One box is used to fit the lines for hands and the other box is used to fit the lines for feet based on the number of the white pixels in each box, as illustrated in Figure 4.6.

4.3 Kinematic features extraction

Once the human skeleton fitting is completed, the centroid angles of the distal segments on the limbs can be detected as features to identify the pose of the distal limb segments. In order to determine the centroid angle, the angles of the endpoints of the distal segment, with respect to the vertical body axis, are averaged for each segment. Therefore, each distal limb segment is described by its centroid angle with respect to the vertical body axis. The five-angle vector which represents the pose of the torso and four distal limb segments is created using these centriod angles. Since all the five-angle measurements are relative to the vertical body axis in one frame, it is necessary to add other information about the motion of the human as function of time to the classifier to discriminate among the actions that have similar angles and different positions such as picking up and jumping. Therefore, Translation Distance TD , which gives the difference in position between two consecutive frames, is added to the descriptor vector to increase the classification accuracy. Consequently, the resulting kinematic features extracted from each frame are five-angles of displacement from the vertical body axis and TD of the position of the hip.

$$TD = \sqrt{(x_{c_t} - x_{c_{t-1}})^2 (y_{c_t} - y_{c_{t-1}})^2} \tag{4.7}$$

where: $(x_{c_t} - x_{c_t})$ is the position of the hip centroid in the current frame.

This yields a six-element descriptor vector with precise orientation and translation data that represent the kinematic features of the human body for each frame, as shown in Figure 4.6.

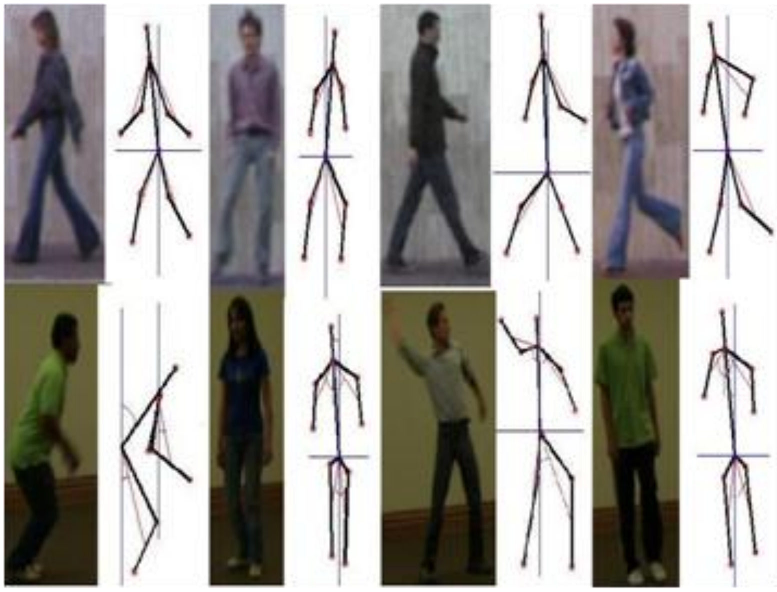


Figure 4.6. Each frame shows the five angles that provide orientation data of torso and limbs.

A simple tracker, using Kalman filter, is applied to the bounding boxes which represent the human body to track their position and predict their location in the next frame. With the tracker the motion direction can be readily estimated using the difference in horizontal position between the centers of the bounding boxes in two consecutive frames. We also assume that the camera is oriented horizontally as indicated in Figure 4.6, therefore the motion is one-dimensional. But the normalization procedure can also be applied to the y-axis to allow for arbitrary camera orientation.

4.4 Action classification

GMM is a parametric probability density function represented as a weighted sum of Gaussian component densities. GMM parameters are estimated from training data using the iterative Expectation Maximization (EM) algorithm from a well-trained prior model. In this dissertation, the Expectation Maximization algorithm [64] was used to train separate GMMs each representing one action using descriptor feature vectors extracted from video sequences of multiple subjects. Given an observation sequence, $X = (x_1, x_2, \ldots, x_M)^T$, and a GMM configuration, the parameters of the GMM, which in some sense best matches the distribution of the training feature vectors, were estimated. Afterwards, the GMMs are utilized to classify the action performed in a given video into different classes such as walking, running, etc.

Let us assume that $\Psi = \{\Psi_1, \ldots, \Psi_z\}$ is a set of GMMs registering the motion patterns of z actions and $\Omega = \{\theta_1, \ldots, \theta_z\}$ contains the label for the corresponding action. In our approach, several sequences of descriptor vectors with six dimensions (five-angles and the translation distance ($d = 6$) was used to compute the distribution probability for N components of GMM $\Psi_k = \{\omega_i, \Psi_i\}_{i=1}^{N}$, which is defined as:

$$f_{\Psi_k}(x) = \sum_{i=1}^{N} p(x|\Omega_i)p(\omega_i) \tag{4.8}$$

where:

- x is a vector with d dimensions , the probability of x over single Gaussian $\Omega_i = \{\mu_i, \Sigma_i\}$ is:

$$p(x|\Omega_i) = \frac{1}{\sqrt{2\pi^2|\sum_i|}} e^{(-\frac{1}{2}(x-\mu_i)^T\Sigma_i^{-1}(x-\mu_i))} \tag{4.9}$$

- μ_i, Σ_i are the mean and variance of Gaussian Ω_i, $\mu_i \in \Re^d$ and $\sum_i$ is $d \times d$ positive matrix, $i = \{1, \ldots, N\}$.

- $p(\omega_i)$ is the weight of i^{th} Gaussian component and $p(\omega_i) > 0$ and $\sum_{i=1}^{N} p(\omega_i) = 1$, $i = \{1, \ldots, N\}$.

- ω_i the weight of the i^{th} GMM component.

After training the GMMs using the EM algorithm, Maximum Log-likelihood method was applied to classify actions in a given video into different classes. Given an observation sequence $X = (x_1, x_2, \ldots, x_M)^T$, $X \in \Re^{M \times d}$, and a GMM configuration, by assuming the observations are independent from each other, then the likelihood function of Gaussian $\Psi_k, (k = 1, \ldots, z)$ can be defined as:

$$\begin{aligned} \zeta(\psi_k) = f_{\psi_k}(x_1, x_2, \ldots, x_M) = \prod_{j=1}^{M} f_{\Psi_k}(x_j) = \\ = \prod_{j=1}^{M} p(x_j|\psi_k) = \prod_{j=1}^{M} (\sum_{i=1}^{N} p(x_j|\Omega_i)p(\omega_i)) \end{aligned} \tag{4.10}$$

The log-likelihood function is calculated by taking the logarithm value of the likelihood function:

$$\zeta^*(\psi_k) = \log \zeta(\psi_k) = \sum_{j=1}^{M} \log(\sum_{i=1}^{N} p(x_j|\Omega_i)p(\omega_i)) \tag{4.11}$$

The maximum log-likelihood criterion is used to classify the action by assigning the maximum observation data sequence of X to GMMs:

$$\hat{\psi} = \arg\max_{\psi_k \in \Psi} \zeta^*(\psi_k) \tag{4.12}$$

where $\hat{\psi}$ denotes the GMM which has the largest log-likelihood for the observation data sequence. The action label corresponding to this GMM is assigned to the observation sequence as follows:

$$Action = \theta_i,\ if(\hat{\psi} = \psi_i), i \in \{1, \ldots, z\} \tag{4.13}$$

4.5 Experiments and results

In order to evaluate the performance of our approach, we evaluated the performance of our approach for human action recognition using three different datasets; DU dataset, Weizmann dataset [3], and KTH dataset [65].

4.5.1 Experiments on DU dataset

The DU dataset, captured at the University of Denver, consists of 305 videos of seven different actions (walking, running, throwing, turning around, jumping, waving, and picking up) performed by seven subjects. The videos were captured using a JVC camcorder at 30 fps with resolution of 640×480. For collecting the videos, each subject was asked to perform each action many times in front of the camera that was fixed at a location inside a building. The actions were captured at an angle at which the camera could view the motion with minimal occlusion. The subjects were given freedom to perform the actions at their own pace at any distance in front of the camera. The number of videos for each action is between four and seven per subject. A few sample frames of the videos, with extracted human skeleton model using our proposed approach, are shown in Figure 4.7.

Leave-One-Subject-Out (LOSO) method was applied to verify the performance of our approach. For all the videos, one subject is removed from the training set and the other subjects are utilized to train separate GMMs for separate actions.

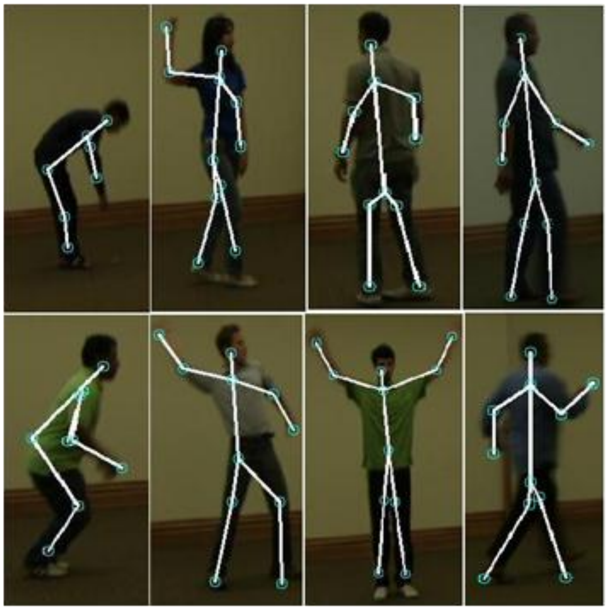


Figure 4.7. shows sample video frames with human skeleton fitting for different actions for DU dataset.

The excluded videos were used to test the accuracy of our approach in classifying the performed actions in videos based on the maximum log-likelihood classification. The test subject was different from the training ones. This process was repeated for all the subjects and all the actions in the dataset separately.

Table 4.1 shows the results of classifying actions in form of confusion matrix for DU dataset based on LOSO. We can observe that the average classification accuracy of our approach is 100% for walking, jumping, and turning, 98% for waving, throwing and picking up actions, and 93% for running which is confused with walking because of the similarity. In all, the average classification accuracy was 98%.

Because of different number of GMM components can influence the classifi-

Table 4.1. Confusion matrix for experiments on the DU dataset.

Actions	Run	Walk	Jump	Turn	Wave	Throw	pick-up
Run	0.93	0.07					
Walk		1					
Jump			1				
Turn				1			
Wave					0.98	0.02	
Throw					0.02	0.98	
pick-up						0.02	0.98

cation result, several GMMs components were considered. In this context, DU dataset was used as validation dataset among the other datasets in order to find the optimum number of GMM components that yields to the best result. Particularly, several GMMs with different number of components (2 to 12) were trained and tested. According to our experiments, the best results were obtained when the number of Gaussian components was equal to six. Therefore, in our experiments with KTH and Wiseman dataset, six Gaussian components were used to evaluate the approach.

4.5.2 Experiments on Weizmann dataset

In another experiment, Weizmann dataset (Blank et al. [3]) was utilized to evaluate the performance of our system for recognizing different actions. Weizmann dataset consists of 90 video sequences (180 × 144), 50 fps frame rate showing nine different subjects, each performing 10 different actions. Weizmann dataset contains images with lower-resolution than DU dataset. It represents silhouettes with varying scales, poses, and directions that make the blob detection more challenging compared with DU data-set. Figure 4.8 shows the results of our human skeleton fitting for sample video frames. Table 4.2 illustrates the confusion matrix for classification experiments on Weizmann dataset based on LOSO. From Table 4.2, we can observe that the accuracy of our approach is 100% for bend, p-jump, side, and wave1, 88% for jack, jump, skip, wave2, and 77% for walk which is confused with side walk and run. The lowest recognition rate (67%) is for run action because of its similarity with other action. The average recognition rate is 90% for all the actions and it is mostly confused by similar action classes, such as skip with run and side walk with walk. If we remove either side walk or run action from the experiments, the average recognition rate will increase to 94%.

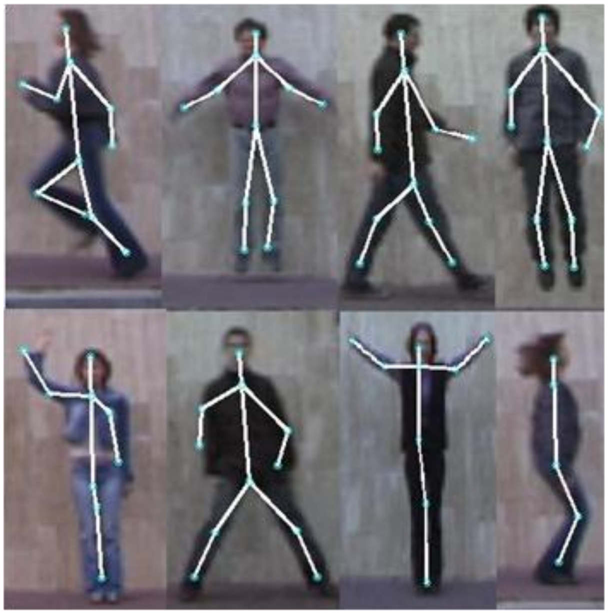


Figure 4.8. shows sample video frames with human skeleton fitting for different actions with low resolution images (Weizmann dataset); from left-top to right-bottom, the actions are:, run, jack, walk, position jump, wave1, side , wave2 and jump.

In fact, there are two main reasons that make the recognition rate on Weizmann dataset less than the recognition rate on DU dataset. First, the number of training samples in Weizmann dataset is limited to one video per actor per action, compared to DU dataset that contains multiple sample videos per action. Second, because of the low resolution images in Weizmann dataset, actors have a smaller size which makes the limb detection process more difficult. In all, our approach was able to process blobs though images that have low resolution.

4.5.3 Experiments on KTH dataset

The third dataset, KTH dataset (Schuldt et al. [65]), contains six types of human actions (walking, jogging, running, boxing, hand waving and hand clapping) performed several times by 25 subjects in four different scenarios: outdoors with scale variation and different view angles, outdoors with different clothes and indoors. All videos were taken over homogeneous backgrounds with a static camera at 25 fps. The sequences have a spatial resolution of 160×120 pixels and had a length of

Table 4.2. Confusion matrix for experiments on Weizmann dataset.

Actions	Bend	Jack	Jump	P.jump	Run	Skip	Side	Walk	Wave1	Wave2
Bend	1									
Jack		0.9					0.1			
Jump			0.9			0.1				
P.jump				1						
Run					0.7	0.3				
Skip					0.1	0.9				
Side							1			
Walk					0.1		0.1	0.8		
Wave1									1	
Wave2		0.1								0.9

four seconds in average. A few sample frames of KTH dataset with human skeleton fitting are shown in Figure 4.9. In this dataset, we focused on two scenarios that represent the subjects with scale variation and outdoors (scenario 2) and with different clothes (scenario 3) compared with the previous datasets. These scenarios are more challenging than the indoors scenario (s1) and outdoors scenario (S4).

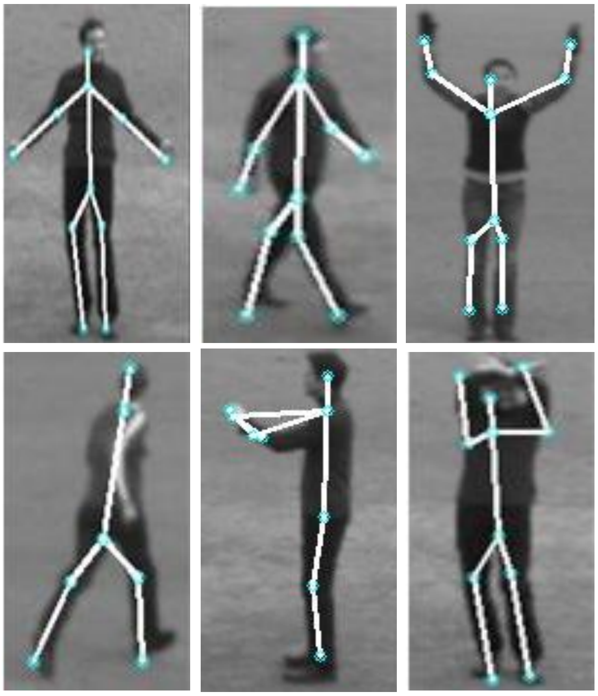


Figure 4.9. shows sample video frames with human skeleton fitting and skeleton for different actions (KTH dataset); from left-top to right-bottom, the actions are: handclap, walk, run, box, and wave.

The confusion Table 4.3 illustrates the classification results in confusion matrix form for our experiments on the KTH dataset (scenario 2). The average recognition rate was 94% for all the actions. Clearly, there is confusion between running, walking, and jogging because of the similarity of these three actions. For instance if the jogging actions is avoided, the average recognition rate increases to 96%. In fact, only seven out of $25 \times 6 = 150$ videos are misclassified in this experiment. Three of the misclassified videos are the Jog action, which are incorrectly labeled as Run action. One video from Clapping action is misclassified as Wave action. The Run and Walk actions are misclassified two times, respectively. Our observation is that these errors only for those actions which have quite a bit of similarity with each other.

Table 4.3. Confusion matrix for experiments on the KTH dataset.

Actions	Run	Walk	Box	clap	Wave	jog
Run	0.9	0.1				
Walk	0.05	0.95				
Box			1			
clap				0.95	0.05	
Wave					1	
jog	0.15					0.85

In order to test the robustness of our method with respect to the variations of actors' clothes, we performed a set of experiments by selecting different subjects with different clothes. We observed that the recognition rate for all the actions did not change with the variations of clothes expect only in one case, when the color of the background and the clothes are the same. This is due to the fact that our segmentation technique depends on the difference in the color between the background and foreground.

We also conducted several experiments to evaluate the robustness of our method with respect to body occlusion and failure in the limb detection. In these experiments we selected a subset of four reference angles instead of five angles. First, the torso angle was removed from the feature vector, and we achieved an average recognition rate of 55.2%. Classification errors were observed in packing up, bending, and jumping actions. In the second run, we removed the right distal limb angle (right arm) from the set and achieved an average recognition rate of 63.1%. In the third run, we removed the left distal limb angle (left leg) from the set and the average recognition rate was 64.7%. These experiments show that the proposed method for activity recognition is sensitive to the occlusion of individuals' body limbs. At the same time, we observed that the recognition rate for some actions(e.g. bending, jumping, and picking up) that highly rely on the torso angle were decreased significantly. But for other actions such as walking and running that involve multiple joint angles (two angles for the legs & two angles for the arms), removing one of these angles has less effect on the overall recognition rate.

Finally, to evaluate the effect of body posture on our approach, we applied our approach to videos of six actions on Weizmann dataset performed in front view or side view with respect to the camera. These actions are Jump, Bend, Wave, Walk, run, skip. As mentioned before, we used only one single view of the camera to construct the nine-segment skeleton model. We observed that in all the cases of front view, the body posture can be described very well by the nine-segment skeleton model even if the arms are close to the body as shown in Figures 4.7-4.9. But in the case of side view, we observed that our approach may have some difficulties in building the nine-segment skeleton model. This is mainly due to the self-occlusion of the body parts and also because the nine-segment skeleton model is constructed in 2D space using only 2D information which really makes estimating the pose of the human body difficult. To overcome such problems, using multiple cameras and also depth sensors such as Microsoft Kinect can be very helpful.

4.5.4 Comparison with another approaches

We compared the performance of our six-element feature descriptor with other relevant action recognition methods. These methods are the skeleton models in [1, 17], the trajectory of the body joint points in [16] , and methods that utilized the Weizmann or KTH datasets [28, 66]. Table 4.4 summarizes the comparison with different approaches based on the reported results in the published papers.

Table 4.4. Comparison with different approaches.

Approach	Dataset	Video Resolution	Frame Rate	No. of Actions	No. of Subjects	R. Rate
Single Star Skeleton;						
Chen et al. 2006 [1]	Private	352 × 240	30	10	5	98%
Our Six-element Descriptor	DU	640 × 480	30	7	7	98%
Variable Star Skeleton;						
Yu and Aggarwal 2009 [17]	Weizmann	180 × 144	50	10	9	93%
3D SIFT;						
Scovanner et al. 2007 [28]	Weizmann	180 × 144	50	10	9	82%
Chaotic invariants;						
Ali et al. 2007 [16]	Weizmann	180 × 144	50	9 no skip	9	92%
Our Six-element Descriptor	Weizmann	180 × 144	50	10	9	90%
Five-layer hierarchical model;						
Ning et al. 2008 [66]	KTH	160 × 120	25	6	25	95%
Our Six-element Descriptor	KTH	160 × 120	25	6	25	94%

The first row of Table 4.4 presents the results obtained by single star skeleton approach which was applied on a private dataset. This method has a 98% recognition rate (similar to our method on DU data-set). Because the images used to evaluate the single star skeleton method had higher resolution compared to other public datasets, then it is hard to have a fair comparison with other approaches that adopted low resolution datasets. Also, the number of videos per subject per action used in [1] was larger (three videos per subject per action) than the number of videos in the KTH and Weizmann datasets (only one video per subject per action) which makes the training process and obviously the recognition rate better. The results of three other methods, variable star skeleton approach [17], 3D SIFT [28], and chaotic invariants approach [16], that utilized the Weizmann dataset are compared with our approach in Table 4.4. Variable star skeleton which detects extremities inside the contour rather than the limbs achieved a good recognition rate 94%. The chaotic invariants approach [16] used a semi supervised joint detection approach for the human body to detect the six points (two hands, two feet, the head, and the belly point) as approximation of body joints. Based on trajectories of these reference joints, they achieved a recognition rate of 92% with no skip action compared with our approach that achieves 94% recognition rate with no skip action. The 3D-SIFT [28] method, which uses the motion features to recognize the action, has an 82% recognition rate that is 8% lower than our method. Using the KTH dataset, Ning et al. [66] proposed a system to search for human actions with a five-layer hierarchical space-time model and achieved a 92% recognition rate. In summary, our six-element descriptor has better or comparable recognition rate than other methods in classifying human actions.

4.6 Summary

This chapter presented an approach for fitting distal limb segments in visual data for accurate human skeletonization. The proposed nine-segment skeleton model is suitable for a wide range of applications from surveillance to human motion analysis and human-computer interaction (HCI). For human action recognition, kinematic features were extracted from the nine-segment human skeleton model. This descriptor was represented by GMMs and utilized for action classification using maximum log-likelihood criterion. Our experiments on three different action datasets showed that our approach is able to detect the limbs with a good accuracy and recognize different actions with comparable or better recognition accuracy than other relevant action classification methods.

Nevertheless, there are some restrictions with our proposed approach for distal limb detection. One limitation is the extraction of human silhouette (i.e., background subtraction). In fact our approach requires the body silhouette to be accurately detected from background in images. Hence, in order to build a robust system, a strong approach for dynamic segmentation independent of environment needs to be developed and be utilized with our approach.

Chapter 5

Surface Representation for Human Action Recognition

This chapter presents an approach for human action recognition based on surface representation of the human body obtained from depth map provided by Kinect sensor. In our work, we utilized spherical harmonics representation to extract 3-D shape features of the body silhouette from depth map. Then, spherical harmonics coefficients are calculated up to a user-desired degree L_{max} to describe the human body surface as 3-D shape features. These coefficients are utilized to represent the 3-D body surface in the frequency domain. Hence, the focus of this chapter will be on "Given surface representation of human body", how do we construct different 3-D shape features that can be used to track the human body and recognize human action using sequence of depth maps.

5.1 Overview of the method

Researchers have explored different approaches for human action recognition based on shape features in the past few decades. Most of the approaches proceed in a top-down fashion where the whole body of person is localized in the region of interest (ROI), which is usually obtained through tracking or background subtraction. Then, the ROI is encoded in order to create the shape features. In general, shape features look at the changes in the whole body shape and try to recognize actions based on dominant motions in the body parts. In this context, two types of methods based on shape features can be used; silhouette-based features and contour-based features where the first method takes into account all the points within a shape and the latter method only extracts the shape boundary (contour).

From the extensive work on silhouette-based features for human action recognition presented in chapter 2, we have observed that silhouettes contain rich shape information of the human body. However, most shape features are carried either by the interesting points on the contour or surface points on the silhouettes. This observation forms the core concept of our proposed method to characterize the human action using the surface points on the silhouette obtained from sequences of depth maps.

In the method presented here, sequence of depth maps which represent the silhouettes of human body are acquired using Kinect sensor. Then, background subtraction is used to segment the body silhouette, that represents the surface points of human body, from the scene. Afterwards, the missing points (holes) are restored using image processing techniques such as closing operation and the image noises (spikes) are reduced using median filtering. After preprocessing, the depth map is adaptively sampled based on the image depth variation to reconstruct a 3-D mesh for further processing. Sample points in depth images are triangulated using the constrained Delaunay triangulation algorithm [67] and projected in 3-D space. Then, spherical harmonics decomposition, which decompose the 3-D mesh structure into a set of spherical harmonics coefficients, is performed.

Spherical harmonics decomposition is defined as a 3-D extension of Fourier transform for representing 3-D shape features using a spherical function. Specifically, spherical harmonics decomposition was originally used as a type of parametric surface representation for radial surfaces [68, 69] and later extended to more general shapes by representing the shape surface using a spherical function $(x(\theta,\phi), y(\theta,\phi), z(\theta,\phi))$ with θ as the polar angle and ϕ as the azimuth angle.

Our 3-D shape features are developed based on spherical harmonics decomposition of depth map for human silhouette. In the spherical harmonics decomposition, first, spherical parametrization [69] which creates a continuous and uniform mapping from the 3-D mesh surface to the surface of a unit sphere is adopted. The result of the spherical parametrization process is a bijective mapping between each vertex point on a 3-D mesh surface and a pair of spherical coordinates, θ and

ϕ. The aim of the spherical parametrization is to parameterize the surface onto a unit sphere. Afterwards, spherical harmonic expansion [70] is applied to the spherical function $(x(\theta,\phi), y(\theta,\phi), z(\theta,\phi)))$ which represents the 3-D mesh surface to calculate a set of spherical harmonics coefficients. These coefficients (features) are used to create the 3-D shape feature to recognize human actions using C-SVM classifier. The accuracy of the 3-D shape feature relies only on the surface points and the spherical harmonics band that is used to determine the number of coefficients to describe 3-D shape feature. Figure 5.1 illustrates a general overview of our proposed method.

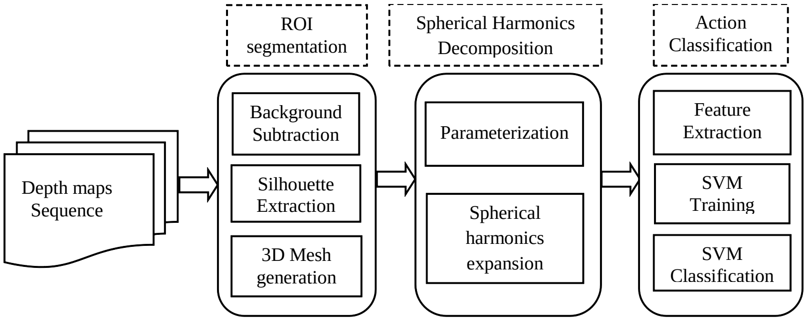


Figure 5.1. Our proposed method for human action recognition using spherical harmonics representation.

5.2 Video Processing for 3-D Shape Features

In this section, we briefly review the main steps of our method and the proposed 3-D shape features based on spherical harmonics representation that serves as the foundation for human action recognition.

5.2.1 Segmentation and 3-D mesh generation

An adaptive Gaussian model using probability density functions for sequence of depth maps were built to evaluate if the pixels belong to background or foreground. In our work, Stauffer et al. approach [59] is employed to segment moving objects from the static background in order to represent the ROI. A median filter and closing operation are implemented to the segmented object to reduce the high frequency noises in the depth map and to restore the missing pixels. After the preprocessing step, the depth map is used as 3-D data source to generate a 3-D mesh surface. In order to reduce the number of mesh points and come up with less number of triangles (for the sake of computation), down-sampling method is used to extract an optimal amount of points from a depth map. After that, a triangular mesh surface is generated from the sampled points using constrained Delaunay triangulation algorithm [67]. In our approach, the external boundary points are used as constraint in the triangulation to keep the connectivity among the silhouette points and to generate an accurate boundary. Figure 5.2 demonstrates depth map samples and generated 3-D mesh surface of two separate actions.

5.2.2 Spherical parametrization

Spherical harmonics was originally used as a type of parametric surface representation for radial surfaces and later extended to more general shapes by representing

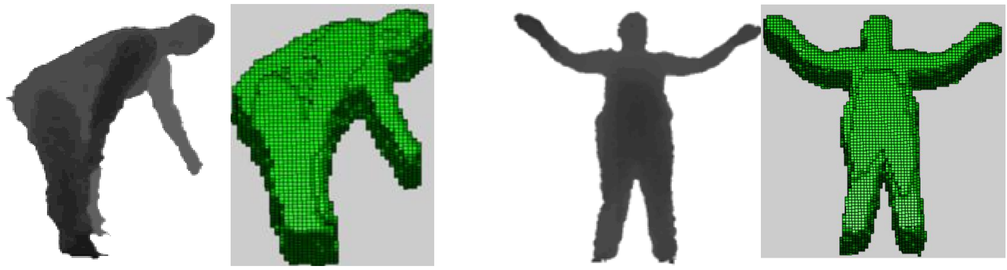


Figure 5.2. Examples of depth map and 3-D mesh surface for samples frames showing the wave and pick-up actions.

the shape surface using spherical function $S(\theta, \phi) = (S_x(\theta, \phi), S_y(\theta, \phi), S_z(\theta, \phi))$ with θ as the polar angle and ϕ as the azimuth angle. Thus, it is convenient to use spherical coordinates (θ, ϕ), as illustrated in Figure 5.3, instead of the Cartesian coordinates. The variable θ is the polar angle with $\theta \in [0, \pi]$ and ϕ is the azimuth angle with $\phi \in [0, 2\pi]$. Radius R represents the distance from the origin of the coordinate system, but it can be omitted for normalized coordinates, which all lie on a unit sphere ($R = 1$) to create a uniform distance from the origin.

Spherical parametrization algorithm [68, 69] is used to establish a one-to-one mapping between 3-D points on human body surface (vertices) and 3-D points on the unit sphere. The result of the spherical parametrization process is a bijective mapping between each 3-D point on a human body surface and a pair of spherical coordinates, θ and ϕ. Therefore, the human body shape is represented as spherical function: $S(\theta, \phi) = (x(\theta, \phi), y(\theta, \phi), z(\theta, \phi))^T$, where the function $S(\theta, \phi)$ specifies the distance from a specified origin point to each point on the surface of the sphere

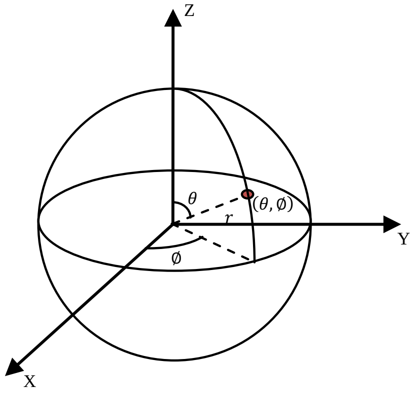


Figure 5.3. Conventions for the spherical coordinates.

for all three coordinates $x = R\cos\phi\sin\theta, y = R\sin\phi\sin\theta, z = R\cos\theta$.

Our method follows Shen et al. work [69], where we adapt the initial mapping method to voxel surface without applying the optimization method in order to accelerate the algorithm and to establish a one-to-one mapping on the unit sphere. In our work, we convert the mesh surface into voxel surface in order to fill all the holes on the surface before the spherical parameterization process.

In order to compute the spherical harmonics coefficients of a given 3-D human surface, that surface needs to be a 3-D connected closed surface with no holes. In practice, this topology requirement may not be satisfied because of the topology of the depth map (2.5-D) and the missing data due to sensor errors. In other words, some surface patches may have holes because of the complexity of the surface or sensor errors. Therefore, to compute the spherical harmonics coefficients for those surfaces, an interpolation strategy is proposed. The idea of the interpolation strategy is to convert 2.5-D into 3-D by closing the back side and fill the holes on the surface resulting from missing data.Since there are no data in the back area, the most direct solution is to fit a plane into the back side based on the boundary of the human body. In this context, the vertices of the boundary are projected onto the plane, then, the connectivity function is used to fill voxels onto the holes where the data is missing. After filling the holes, the spherical harmonics coefficients for the 3-D connected closed surface of the whole body can be computed using spherical harmonics decomposition. Figure 5.4 shows an example of 2.5-D surface and the 3-D closed surface before and after closing the surface.

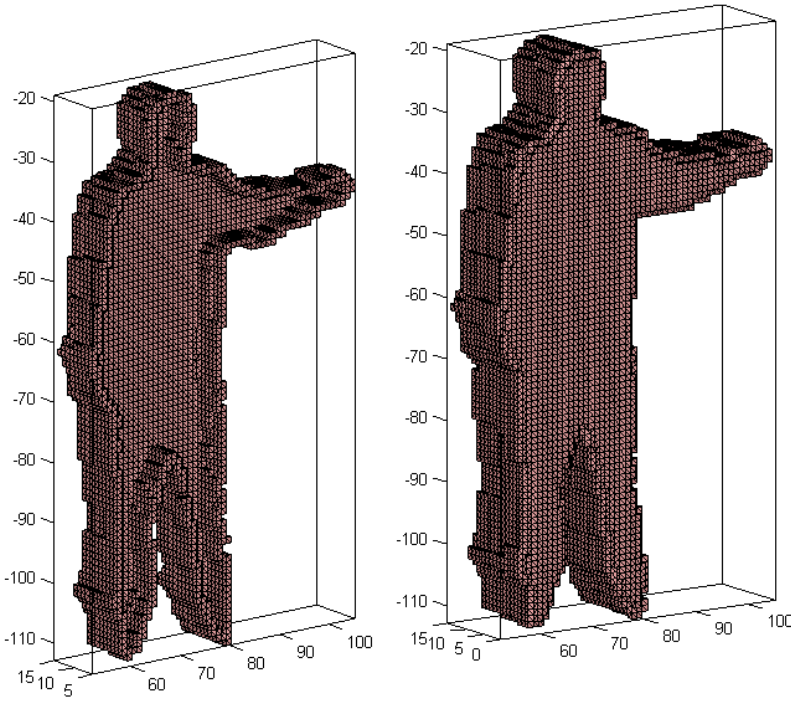


Figure 5.4. Example of 2.5-D surface and the 3-D closed surface before and after closing the surface.

5.2.2.1 Parametrization algorithm

Parameterizing a 3-D surface over a unit sphere is equivalent to mapping its connectivity surface onto a unit sphere. Specifically, a 3-D surface is mapped onto a unit sphere under a bijective mapping (i.e. every point on the surface has to map to one point on the sphere). In this context, a spherical function is created by mapping the voxel surface (vertices) to the unit sphere. Our approach follows the work of Brechbuhler et al. [68] and adapts his initial mapping method to voxel surface without applying local and global smoothing steps. A more detailed discussion of parametrization and the optimization algorithm can be found in [68, 69]. Figure 5.5 shows a human body surface and its mapping onto a unit sphere.

Under this framework, a quadrate-based surface (voxel surface) serves as input for the parameterization algorithm used to preserve and minimize the area and topology distortion. The parametrization process is implemented in polar coordinates, where the two polar coordinates θ and ϕ are determined for all Cartesian coordinates (x, y, z) of vertices in two separate steps. First, with latitude θ two vertices with maximum distance are identified in the voxel surface; these two vertices have to be selected as poles for parametrization process. Then, a Laplace equation $\bigtriangledown^2\theta = 0$ (except at the poles),with Dirichlet conditions $\theta_{north} = 0$ and $\theta_{south} = \pi$ is solved for latitude θ to assign a latitude value for every vertex. The Laplace equation is approximated by finite second differences of the available direct neighbours for every vertex's latitude (except the poles). These conditions

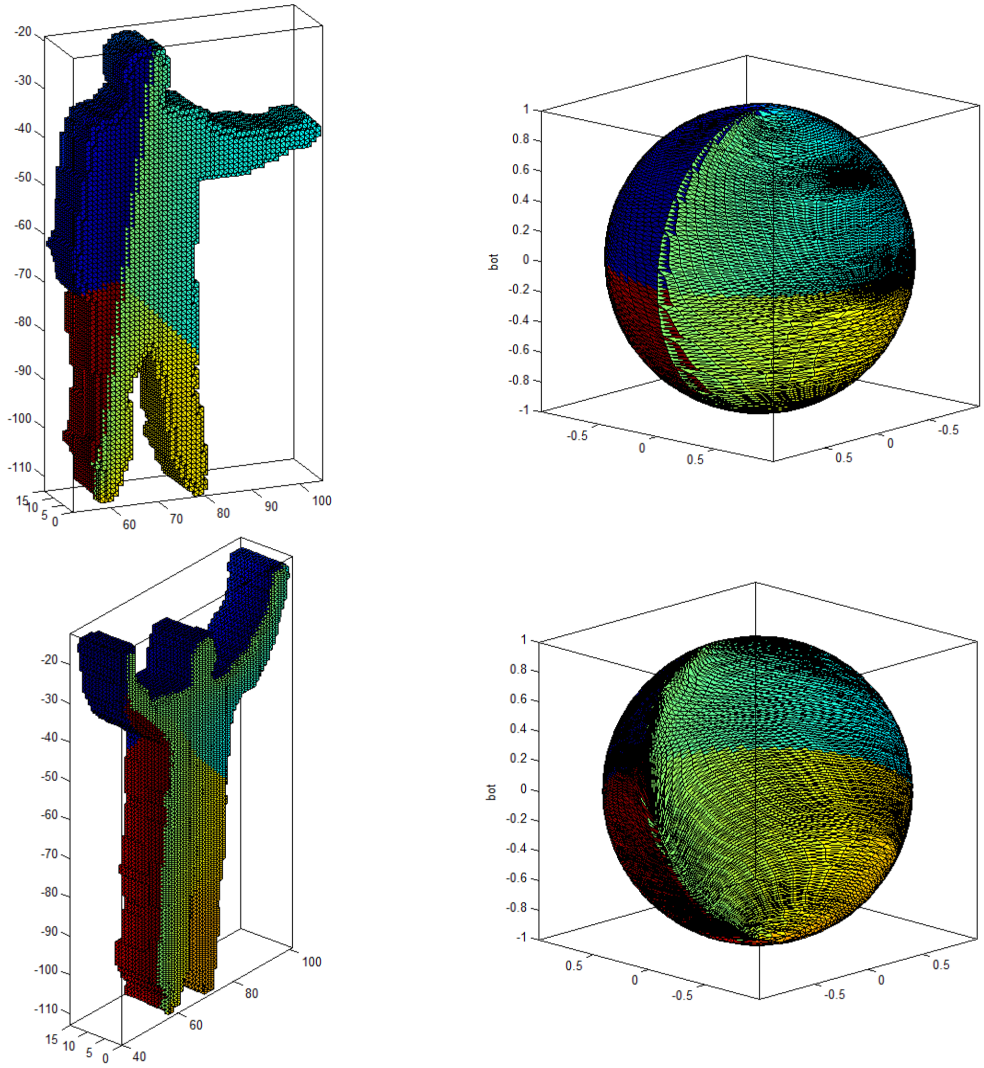


Figure 5.5. Examples of the voxel surface and it is spherical parametrization.

form a sparse set of linear equations, which can be written in the form $A\theta = B$, where A is a $n \times n$ matrix (direct neighbours for every vertex), n is the vertices number, $\theta = \{\theta_0, \theta_1, ..., \theta_n\}$, and B is a vector of constants. The solution for this sparse symmetric linear system of equations 5.1 has the property that latitude θ varies monotonically between the poles.

Algorithm 1: INITIAL PARAMETERIZATION ALGORITHM FOR LONGITUDE θ.

```
Input: Given voxel surface $V_{(x,y,z)}$ as a quadrate-based surface mesh.
Output: Spherical function $S(\theta,\phi)$ presents the object surface on a unit
        sphere.
Checking surface connectivity with Voxel surface $V_{(x,y,z)}$:
$n \leftarrow$ number of vertices
Set up matrix A:
for vertex $\leftarrow 1$ to $n$ do
    neighbours := number of direct neighbours;
    for $M_{(x,y,z)} \leftarrow 1$ to neighbours do
        if $M_{(x,y,z)} \neq pole$ then
            $C_{vertex} := -1$;
    $C_{vertex,neighbours} :=$ average $C_{vertex}$;
    $A_{vertex,vertex} = C_{vertex,neighbours}$
Set up Vector B:
set all entries of B to 0
for vertex $\leftarrow 1$ to $n$ do
    neighbours := direct neighbours of south pole;
    $B(neighbours) = \pi$
Calculate the latitude $\theta = \{\theta_0, \theta_1, ..., \theta_n\}$ by solving linear system $A\theta = B$
return latitude vector $\theta$
```

$$\begin{pmatrix} B_1(1,1) \\ B_2(2,2) \\ B_3(3,3) \\ \vdots \\ B_n(n,n) \end{pmatrix} = \begin{bmatrix} A(1,1) & A(1,2) & A(1,3) & \dots & A(1,n) \\ A(2,1) & A(2,2) & A(2,3) & \dots & A(2,n) \\ A(3,1) & A(3,2) & A(3,3) & \dots & A(3,n) \\ \vdots & \vdots & \vdots & \dots & \vdots \\ A(n,1) & A(n,2) & A(n,3) & \dots & A(n,n) \end{bmatrix} \begin{pmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \vdots \\ \theta_n \end{pmatrix} \tag{5.1}$$

The following portions of pseudo-code shows how to set up the matrix A and generates a right hand side vector in order to calculate the latitude $\theta = \{\theta_0, \theta_1, ..., \theta_n\}$ by solving linear system.

Unlike latitude, longitude is a cyclic parameter. Thus, a date line is introduced as discontinuous line running from pole to pole, and the step height is 2π. The date line is chosen as a path with steepest latitude ascent in each of its vertices. The values crossing the date line from west to east are decremented by 2π, while values propagated to the west are incremented by 2π. Also, the poles and all links to them are removed from the net, making the topology of the net without poles. Then, the cyclic Laplace equation $\bigtriangledown^2\phi = 0$ (with date line) is solved, as was done with the equation for latitude. The new system of linear equations is structurally identical to the one for latitude. Therefore, a latitude θ and a longitude ϕ are computed for every vertex to define a continuous, unique mapping from the surface of the original object to the surface of a sphere. A more detailed discussion of the initial parameterization algorithm can be found in [68].

5.2.3 Spherical harmonics decomposition

After spherical parametrization, the spherical harmonic expansion [70] can be used to compose the human surface into a complete set of spherical harmonics basis functions and coefficients. Spherical harmonic expansion is essentially a Fourier transform technique that defines a 3-D surface using a spherical function $S(\theta, \phi)$ and transforms them into a set of spherical harmonics coefficients $(c_{lx}^m, c_{ly}^m, c_{lz}^m)^T$ in the frequency domain. Theses coefficients can be calculated up to a user-desired band L_{max} by solving a system of linear equations.

The spherical function can be transformed in terms of spherical harmonics functions based on Brechbuhler et al. [68, 69] approach. They presented the surface with an extended spherical harmonics which can model any simply connected 3-D object for all spherical coordinates.

Thus, given the spherical function $S(\theta, \phi)$ defined on the surface of the unit sphere and the orthogonal spherical harmonics bases Y_l^m, where Y_l^m denotes the spherical harmonic of degree l and order m:

$$Y_l^m(\theta, \phi) = \sqrt[2]{\left[\frac{(2l+1)(l-m)!}{(4\pi(l+m)!)}\right]} P_l^m(cos\theta)e^{(im\phi)} \tag{5.2}$$

$P_l^m(x)$ is the associated Legendre polynomial defined by the differential equation:

$$P_l^m(x) = \frac{(-1)^m}{(2^l l!)}(1-x^2)^{\frac{m}{2}} \frac{d^{l+m}}{dx^{l+m}}(x^2-1)^l \tag{5.3}$$

L_{max} is the band-width of the spherical harmonics, l and m are integers with $0 \leq l < L_{max}$ and $|m| \leq l$.

Then, the spherical harmonics coefficients $c_l^m = (c_{lx}^m, c_{ly}^m, c_{lz}^m)$ which are related to $S(\theta, \phi)$ can be calculated using the following equation:

$$S(\theta, \phi) = \sum_{l=0}^{L_{max}} \sum_{m=-l}^{l} c_l^m Y_l^m(\theta, \phi) \tag{5.4}$$

These complex-valued coefficients, c_l^m, can also independently be decomposed in terms of the spherical harmonics as:

$$x(\theta, \phi) = \sum_{l=0}^{L_{max}} \sum_{m=-l}^{l} c_{lx}^{m} Y_l^m(\theta, \phi) \tag{5.5}$$

$$y(\theta, \phi) = \sum_{l=0}^{L_{max}} \sum_{m=-l}^{l} c_{ly}^{m} Y_l^m(\theta, \phi) \tag{5.6}$$

$$z(\theta, \phi) = \sum_{l=0}^{L_{max}} \sum_{m=-l}^{l} c_{lz}^{m} Y_l^m(\theta, \phi) \tag{5.7}$$

Clearly, calculating the spherical harmonics coefficients linearly depends on the spherical harmonics band L_{max} and number of surface points. In particular, these coefficients are 3-D vectors which are computed up to a user specified maximum band to construct the 3-D shape features. Therefore, based on spherical harmonics decomposition, we can extract our 3-D shape features with different frequency harmonics. This 3-D shape features defines the body shape in one frame. In order to represent an action that includes t number of frames, our 3-D spatio-temporal feature for all the frames are concatenated together to build a single spatio-temporal feature vector for action description.

In our approach, spherical harmonics coefficients are determined using standard least square technique [71]. The computational complexity of solving a standard least square estimation using a Moore-Penrose algorithm for matrix inversion is $O(NM2) + O(M3)$ where N is the number of equations (surface points) and M is the number of unknowns (number of coefficients,$(M = L_{max} + 1)^2$.

5.2.4 3-D Shape features

A 3-D shape features (S_F) is an abstraction of the human body representation by capturing shape information in a structure that is well-suited for comparison. Therefore, based on spherical harmonics decomposition, we can build 3-D shape features with different frequency harmonics (f_l) up to a user maximum band L_{max} :

$$S_F = \{f_0, f_1, , f_2, ...f_{L_{max}}\} \tag{5.8}$$

where f_l is defined as:

$$f_l = \{c_l^m, m = -l, ..., l\} \tag{5.9}$$

By combining these two last Equations, and taking the real value for each spherical harmonics coefficients in 3-D space, we create the 3-D shape features. This shape feature defines the body shape in one frame. In order to represent an action that includes t number of frames, our shape descriptor for all the frames are concatenated together to build a feature vector for action description.

5.3 Action classification

In our approach, a canonical C-SVM classifier with a single kernel function and a group of kernel parameters were used to classify the action based only on the shape features. The canonical C-SVM [72] is a discriminative classifier for binary and non separable classification problems.

Let's assume we are given a set of N training samples $\{(x_i, y_i)\}_{i=1}^{N}$, where x_i is the i^{th} feature vector of the training set $\mathcal{X}$ with dimension D, and $y_i \in \{-1, +1\}$ is its corresponding class label. By solving the optimization problem defined in Equation 5.10, SVM is able to find a linear discriminative hyperplane with the maximum margin between samples from different classes in the feature space $\mathbb{R}^D$. The hyperplane is characterized by its direction w and its exact in-space position b as shown in Figure 3.2.

$$\begin{aligned} \min_{w,b,\xi} \quad & J(w,b,\xi) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i \\ s.t. \quad & y_i(w^T\phi(x_i) + b) \geq 1 - \xi_i, \quad i = 1, 2, ..., N \\ & \xi_i \geq 0, \quad i = 1, 2, ..., N \end{aligned} \tag{5.10}$$

where $\xi = (\xi_1, \xi_2, ..., \xi_N)^T$ is known as the vector of slack variables, and C is a positive constant value preset to control the relative influence of non separable samples.

In the canonical C-SVM, solving the optimization problem defined in Equation

5.10 is usually pursued by using its dual form:

$$
\begin{aligned}
\max_{\alpha} \quad & L(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j k(x_i, x_j) \\
s.t. \quad & 0 \leq \alpha_i \leq C, \sum_{i=1}^{N} \alpha_i y_i = 0
\end{aligned}
\tag{5.11}
$$

where $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_N)^T$ is the vector of Lagrangian dual variables corresponding to each feature vector, $k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle$ is a kernel function which maps two D-dimensional features to be a positive scalar, and $\phi(\cdot)$ is a map that maps the feature space $\mathcal{X}$ into a projected space $\mathcal{H}$. Therefore, once given a test sample $x_0 \in \mathbb{R}^D$, the label of x_0 (denoted by y_0) can be calculated according to the following function:

$$
y_0 = sgn(w^T \phi(x_0) + b) = sgn(\textstyle\sum_{i=1}^{N} \alpha_i y_i k(x_i, x_0) + b) \tag{5.12}
$$

where $sgn(\cdot)$ is the sign function.

5.4 Experimental results

In order to evaluate the performance of the proposed 3-D shape feature, We used the MSR-Action 3-D dataset [11] for recognizing human actions performed by different subjects. MSR-Action 3-D dataset [11] is an action dataset of depth map sequences captured by a depth camera (RGBD sensor). This dataset contains different human actions: *high arm wave, horizontal arm wave, hammer, hand catch, forward punch, high throw, draw x, draw tick, draw circle, hand clap, two hand wave, side boxing, bend, forward kick, side kick, jogging, tennis swing, tennis serve, golf swing, pick up and throw.* It includes twenty actions performed by ten subjects. Each action was performed 2 or 3 times by each subject. The size of the depth map is 320 × 240. All the subjects were facing the camera during the performance. The subjects were given a freedom to perform the actions at their own place in front of the camera. Furthermore, most of the actions in this dataset involve the movement of limbs (i.e. arm, leg) in one place, which makes most of the actions highly similar to each other. Figure 5.6 demonstrates depth map samples of five separate actions after segmentation. In fact, this dataset is more challenging compared with the previous datasets such as Weizmann and KTH datasets in human action recognition.

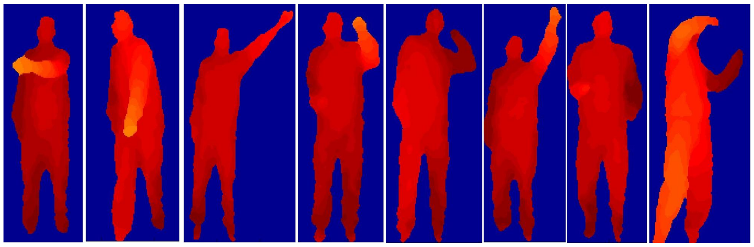


Figure 5.6. Sample depth maps for horizontal arm wave, hammer, high throw, draw circle, throw, hand catch, jogging and tennis swing actions.

To perform action classification using the proposed 3-D shape feature, we used LibSVM [39] classifier with single kernel function. First, we trained several SVMs for different actions using Leave-One-Subject-Out (LOSO) method. For all the videos, one subject was removed from the training set and the other subjects were utilized to train separate SVMs for separate actions. Then, the excluded videos were used to test the accuracy of our approach in classifying the performed actions in videos. The test subject was different from the training ones. This process was repeated for all the subjects and all the actions in the dataset separately. We tested different kernel functions such as linear function and Radial Basis Function (RBF) where RBF kernel showed better best results, As result we used SVM classifiers with RBF kernel.

Figure 5.7 illustrates the results of classifying actions in form of confusion matrix. From the Figure 5.7, we can observe that the average recognition rate of our approach is 74.1% for all the actions. The similar actions are confused by each other. We believe this miss classification to be due the fact that our 3-D shape feature does not consider the motion and translation (i.e., it is invariant to object translation) which makes it difficult to recognize the action from shape structure only. Therefore, extra motion information needs to be included in the feature level to enhance the classification performance. In this context, we need to develop a method that can be used to fuse shape feature and motion feature in the feature level to enhance the recognition rate of human activity recognition. For instance, the motion features of human limbs, such as forearms and shins, may be augmented with the shape features that describe the silhouette in order to improve the pattern classification performance.

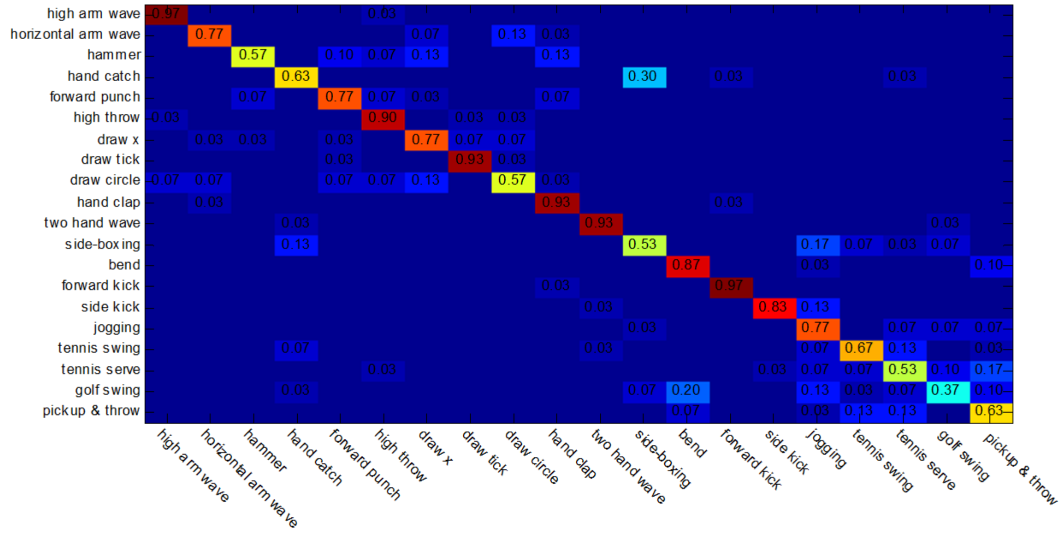


Figure 5.7. Confusion matrix using spherical harmonics coefficients as shape feature with single kernel function (C-SVM Classifier).

5.5 Summary

This chapter presents a study on recognizing human actions using surface representation of depth map. We have utilized the concept of spherical harmonics representation to create 3-D shape feature. Thus, a 3-D shape feature is calculated from the surface representation of body silhouette using spherical harmonic decomposition. Background subtraction, followed by spherical parametrization, was used to describe the depth maps of human body by 3-D points. Afterwards, the 3-D points were mapped onto a unit-radius sphere and then decomposed into a set of spherical harmonic coefficients (similar to Fourier transform). The real values of these spherical harmonic coefficients were used to create a 3-D shape feature.

The proposed 3-D shape feature is used to describe human actions in a sequence of depth maps. The actions were then automatically classified using C-SVM classifier. We conducted experiments on twenty human actions performed by ten subjects. Experimental results have clearly shown the promising performance of the proposed method and also the advantages of using 3-D points over the 2-D silhouettes. Therefore, the contribution of our method is summarized as follows; new approach for human activity recognition based on a set of spherical harmonic representation. In fact, by representing the human body in Fourier domain, the spherical harmonics coefficients can describe the whole body shape as set of shape features. These features are used to create the 3-D shape descriptor which is invariant to scaling and translation.

Chapter 6

Multi-Features for Human Action Recognition

In this chapter, we present multi-modal approach using surface representation and kinematic structure to characterize the human activities. Using the depth information provided by the Kinect sensor, a set of shape features and motion features are extracted to classify the human activities. The shape features are used to describe the human body shape using spherical harmonics representation. The other features are the motion of the 3-D joint positions (i.e. the end points of the distal limb segments) of the human body. These two sets of features are fused using MKL technique for human action recognition. We evaluated and compared our proposed method with the state-of-the-art methods using videos of two challenging human activity datasets. The results of our experiments show promising results and a good direction in incorporating different types of features using MKL technique.

6.1 Overview of the method

In the past decades, significant amount of researches have been done in the field of human action recognition in visual data using sequence of 2-D images. Most of these researches are based either on shape features or motion features. Recently, researchers has paid more attention to use 3-D spatio-temporal features for describing and recognizing human actions [12, 7, 33, 34, 36] due to easy access to depth information (e.g. Kinect sensor).

In general, 3-D spatio-temporal features look at the changes in the body shape based on dominant motions in the human limbs [12, 33]. With the 3-D spatio-temporal features, the changes in the body shape can be detected and represented as space-time volumes. These features mainly focus on the representation of the shape and motion as a function of time. Therefore, the main idea of these methods is to recognize human action by detecting/describing the changes in the human limbs either by detecting the 3-D motion of human limbs or through measuring similarities among different 3-D space-time volumes.

Most of the early methods for human action recognition concentrated on sequence of RGB data captured by regular cameras. This is due to the fact that 2-D images are easier to acquire. In addition, surveillance and security systems utilize regular 2-D cameras for video acquisition. Recently, the developed commodity depth sensors such as Kinect [10] have opened up new possibilities of dealing with

3-D data. This type of sensor has given the computer vision community the opportunity to acquire RGB images as well as depth maps simultaneously at a good frame rate (30 fps) with a good resolution. The depth maps are able to provide additional 3-D data information that can help to distinguish different silhouettes of human body shapes. As we can see in Figure 6.1, the depth map provides additional information as 3-D data which is expected to be helpful in distinguishing poses of silhouettes. Furthermore, comparing with RGB image, the depth map increases the amount of information that can help to detect the 3-D joint positions.

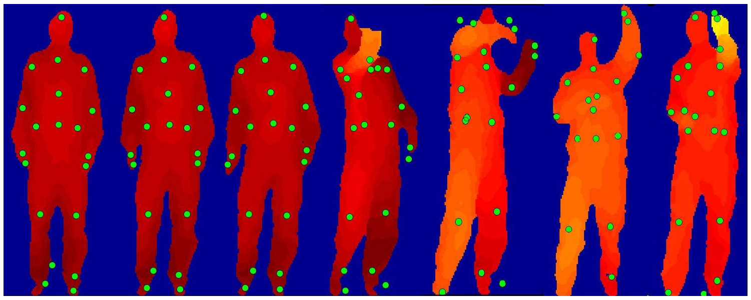


Figure 6.1. Sample of depth maps sequence with 3-D joint positions for Tennis Serve action.

The research in human action recognition based on sequence of depth maps has motivated with the release of the Kinect for windows SDK, which utilized to estimate the 3-D joint positions. Although RGBD sensors produce a better quality of 3-D motion than those estimated from RGB sensors (e.g., Stereo vision systems for 3-D estimation), the estimated 3-D joint positions are still noisy and fails when there are occlusions between human limbs such as two limbs crossing each other. In addition, the motion of 3-D joint positions alone is insufficient to distinguish similar activities such as ”drinking” and ”eating”. Therefore, extra information needs to be included in the feature level to enhance the classification performance. In this context, we need to develop a method that can be used to fuse different types of features in order to create a new feature set from multiple feature sets to represent human action. For instance, the motion features of human limbs, such as forearms and shins, may be augmented with the shape features that describe the silhouette in order to create a new feature set that can help to improve the pattern classification performance.

Feature fusion methods are widely applied in human action recognition due to the fact that it is often insufficient for a single type of feature to represent variations among multiple motions of human body. For lower level fusion as presented in [73, 74], different types of features are combined at the vector level, and the fused feature vector is sent to a single classifier for action recognition. Whereas, for higher level feature fusion as proposed in [75, 76], multiple classifiers are designed for human action labelling from different types of features, and the final decision is made by taking into account the complementaries among classifiers.

Most recently, multikernel learning (MKL) techniques as surveyed in [51] have been proposed for feature fusion at the kernel level within kernel-based classifiers. The works presented in [48, 50] show that MKL can enhance the discrimination power and improve the performance of classifiers during human action recognition. The idea behind MKL is to optimally combine different kernel matrices calculated based on multiple features with multiple kernel functions. The authors of [51] justified the superiority of MKL-based SVM over canonical SVM by showing that the learned discriminant hyperplane of MKL-based SVM performs better than canonical SVM.

This chapter presents a method to recognize human activities using a sequence of depth maps. The basic idea of our method is illustrated in Figure 6.2. Based on surface representation and kinematic structure, we propose a novel features that can characterize the shape and the motion of human body. In our approach, the shape features, which are extracted from the depth map using spherical harmonics representation, are used to describe the 3-D silhouette structure. While, the motion features, which are extracted from the estimated 3-D joint positions, are used to describe the movement of the distal limb segments of human body. We adopted the distal limb segments in our method because we believe that the distal limb segments, such as forearms and shins, provide sufficient and compact information for human action recognition. Therefore, each distal limb segment is described by the orientation and translation with respect to the initial frame in order to create a motion features. Our approach fuses the shape features with the motion features using SimpleMKL algorithm [54] in order to produce an optimally combined kernel matrix within Support Vector Machines (SVM) for action classification. The kernel matrix has more discriminate power than single feature type based ones due to the utility of multiple features with different kernel functions.

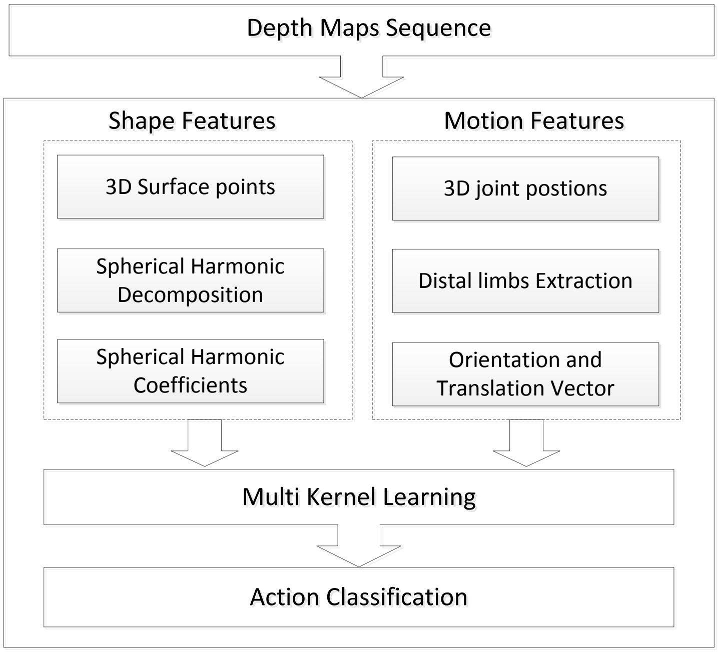


Figure 6.2. Our proposed method for human action recognition based on shape and motion features.

In summary, the main points of this chapter are: 1) Presenting 3-D spatio-temporal features based on spherical harmonics representation for human action recognition. In fact, by representing the depth map of the silhouette in the spherical harmonics domain, the spherical harmonics coefficients can describe the human body shape as a set of shape features in the frequency domain. 2) Presenting 3-D motion features of the distal limb segments in the spatial domain. These different types of features are fused in one kernel matrix in order to enhance the classification performance. 3) The proposed features which are based on depth map are capable to characterize the human-object interactions. These features are evaluated on two datasets: MSRAction3-D dataset [11] and Dailyaction3-D dataset [12]. Our experimental results show promising direction in incorporating different types of features using MKL technique.

6.2 Spatio-temporal features

This section gives a detailed description of two types of 3-D spatio-temporal features that we utilized to classify actions. These features are used to characterize the human limbs motion and the human body shape as a function of time.

In our work, we utilized spherical harmonics representation to extract 3-D shape features of the silhouette, and the kinematic structure of human body to detect the end points of the distal limb segments which are utilized to extract the 3-D motion features. These features are fused using SimpleMKL algorithm [54] with different kernel functions where both the kernel combination weights and discriminate hyper planes of multiclass-SVMs are optimized. In the end, human action classification based on the learnt kernel weights and discriminated hyper planes of multiclass-SVM are used to address the issues of recognizing complex activities. The general framework of our proposed method is demonstrated in Figure 6.2.

6.2.1 Motion features

The distal limb segments (four limbs), as illustrated in Figure 6.3, are utilized to extract the motion features. The motion features represented the orientation and the translation distance of the distal limb segments. In fact, the positions of the distal limb segments are employed to extract the motion features of the human limbs. Our key observation is that the change in positions of the distal limb segments can be utilized to represent the human body movement as discriminative features. This is due to the fact that the distal limb segments provide sufficient information for human limbs movement.

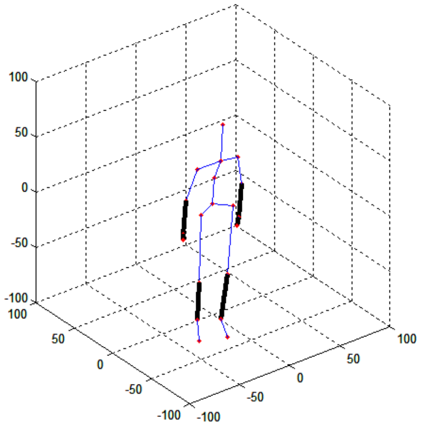


Figure 6.3. The 3-D distal limb segments (four limbs) are used to extract the motion features.

We employed 3-D joint positions of the distal limb segments to characterize the motion features including positions, orientation, and translation. Thus, each distal limb segment L_k is described by its end points as 3-D vector with respect to the body center (hip) in order to create the 3-D unit vector which represents the orientation of the distal limb segment. The orientation of the 3-D vector is defined by a unit vector $\vec{U}_{ab}$ of frame F_t. Therefore, the 3-D coordinates of end points $J(t)$ of distal segments L_k in frame F_t are $J_i^{L_k}(t) = \{j_0, j_1\}$, where $j_a = (x_a, y_a, z_a)$ and k is a distal limb number.

$$\vec{V}_{ab}(t) = \{J_b^{L_k}(t) - J_a^{L_k}(t) \mid a, b = 1, 2, \ldots, N; a \neq b\} \tag{6.1}$$

$$\vec{V}_{ab}(t) = (x_b - x_a)\hat{\imath} + (y_b - y_a)\hat{\jmath} + (z_b - z_a)\hat{k} \tag{6.2}$$

$$\vec{U}_{ab}(t) = \frac{\vec{V}_{ab}(t)}{\|\vec{V}_{ab}(t)\|} = A_x\hat{\imath} + A_y\hat{\jmath} + A_z\hat{k} \tag{6.3}$$

Since all the measurements are relative to the center of the body in one frame, it is necessary to add other information about the motion of the human limbs between the current frame F_t and the initial frame F_{t_0}, which tends to the neutral human pose. Thus, the translation, which gives the difference in position for the distal limb segments between two frames, is computed in order to create a 3-D spatio-temporal feature, and to make the classifier discriminates between the actions that have similar orientation but different positions. Practically, each human subject has four distal limb segments which are tracked by the skeleton tracker [32], each distal limb L_i is represented by 3-D unit vector $\vec{U}_{ab}$ and translation vector D_{tt_0} in each frame.

$$D_{tt_0} = \{\|J_a^t - J_a^{t_0}\| \mid J_a^t \epsilon F_t; J_a^{t_0} \epsilon F_{t_0}\} \tag{6.4}$$

Consequently, the features extracted from each frame is a 3-D unit vector which represent the orientation and translation vector with respect to the initial frame F_{t_0} for each distal limbs as motion features. The combination of these vectors for each distal limb segments forms the motion features representation for each frame. However, all the vectors which represent the 3-D distal limb segments were normalized to reduce intra-class variations among subjects and to be invariant to the body size. We use linear normalization scheme to scale the features in the range $[-1 \text{ to } +1]$. With the tracker [32] the motion direction can be readily estimated using the difference in horizontal position between the centers of the bodies in two consecutive frames. These 3-D motion features define the movement of distal limbs in one frame. In order to represent the motion that includes t number of frames, our motion features for all the frames are concatenated together in order to build a spatio-temporal features vector for motion description. This yields to a spatio-temporal feature with precise orientation and translation data for each human limb in all the frames.

6.2.2 Shape features

Due to the motion similarity in some actions, it is insufficient to only use the motion features to fully describe and model an action. Thus, we create another feature that can describe the human body shape in order to increase the accuracy of the classifier. We employed spherical harmonics coefficients, as shape features, for representation and recognition of human actions from sequence of depth maps.

These features are developed based on spherical harmonics representation of 3-D silhouette. In our approach, the body silhouette is described by 3-D geometric points obtained from the depth map. To reduce the number of mesh points and come up with less number of triangles, depth map points are down-sampled. Then, spherical harmonics decomposition is applied to these points and a set of spherical harmonics coefficients are extracted. These coefficients are used to build spatio-temporal features that describe the human body shape in the spherical harmonics domain as described in chapter 5.

6.2.3 Multi-feature fusion using SimpleMKL

Let's assume we are given a set of N training samples $\{(x_i, z_i, y_i)\}_{i=1}^{N}$, where x_i is a feature vector of the i^{th} sample in the training set $\mathcal{X}$ with dimension D_1 while $z_i \in R^{D_2}$ is another feature vector of that samples (training samples) in the set $\mathcal{Z}$, and $y_i \in \{-1, +1\}$ is their corresponding class label. $k(\cdot, \cdot)$ is a kernel function that maps two feature vectors to be a positive scalar. The SimpleMKL algorithm was proposed to address the MKL base SVM problem by solving the convex problem defined in Equation 6.5.

$$\begin{aligned} \min_{d} \max_{\beta} \quad & L(d,\beta) = -\frac{1}{2}\beta^T(\sum_{m=1}^{M} d_m K^m)\beta + y^T\beta \\ s.t. \quad & \sum_{i=1}^{N} \beta_m = 0, \quad 0 \leq \beta_i y_i \leq C, \quad i = 1,2,...,N \\ & \sum_{i=1}^{M} d_m = 1, \quad d_m \geq 0, \quad m = 1,2,...,M \end{aligned} \tag{6.5}$$

Here, $y = (y_1, y_2, ..., y_N)^T$ is class labels, and $\beta = (\beta_1, \beta_2, ..., \beta_N)^T$ is known as the vector of Lagrangian coefficients in SVM. The multikernel matrix K is generated as $K = \sum_{m=1}^{M} d_m K^m$, where K^m is the kernel matrix calculated based on single type of features with single kernel function. $d = (d_1, d_2, ..., d_M)^T$ is the kernel combination weight vector, and M is the total number of kernel functions in use.

In our work, suppose we are given a pair of samples (e.g., the i^{th} and j^{th}), the fusion of extracted shape ($\{x_i, x_j\}$) and motion features ($\{z_i, z_j\}$) at kernel level within the SimpleMKL framework is handled as follows,

$$K_{i,j} = \sum_{m=1}^{M} (d_m k_m(x_i, x_j) + d_{m+M} k_m(z_i, z_j)) \tag{6.6}$$

Hence in this fusion work, the dimensionality of kernel combination weight vector d is $2M$.

6.3 Experimental results

We choose two challenging public datasets which are known as MSR-Action 3D dataset [11] and MSR-Daily Activity 3D dataset [12] for human activity recognition to evaluate our proposed method. In our experiments, we used spherical harmonics representation, with spherical harmonics band $L = 12$, to represent the shape features, and the movement of distal limb segments as motion features. For each feature two kernel functions were used; Gaussian function and polynomial function, each with different parameters are linearly combined to classify the action using multiclass-SVM classifier. We used the following configuration for kernel functions to fuse multiple types of features with different kernel function parameters based on SimpleMKL framework as defined in the following equations:

$$k_{Gaussian}(x, y) = e^{-\dfrac{\| x - y \|^2}{\sigma^2}} \tag{6.7}$$

$$k_{poly}(x, y) = \langle x, y \rangle^d, d \in \mathbb{N} \tag{6.8}$$

where σ is kernel parameters of Gaussian function and d is the parameter for varying the order of polynomial function. In our experiments, we set different values for parameters of the above two kernel functions with the criterion that they fill a proper range of the defined domain. For Gaussian function, we set $\sigma^2 \in$

$\{0.01, 0.1, 1, 5, 10, 60, 500\}$, for the polynomial function, we set $d \in \{1, 2, 3\}$. Thus, we obtained 10 alternatives for parameterizing the two defined kernel functions (i.e., $7 + 3 = 10$). Hence, given any pair of samples (e.g., the i^{th} and j^{th}), the fusion of extracted shape features ($\{x_i, x_j\}$) and motion features ($\{z_i, z_j\}$) at the kernel level within our framework is handled as follows:

$$K_{i,j} = \sum_{m=1}^{10} [d_m k_m(x_i, x_j) + d_{m+10} k_m(z_i, z_j)] \tag{6.9}$$

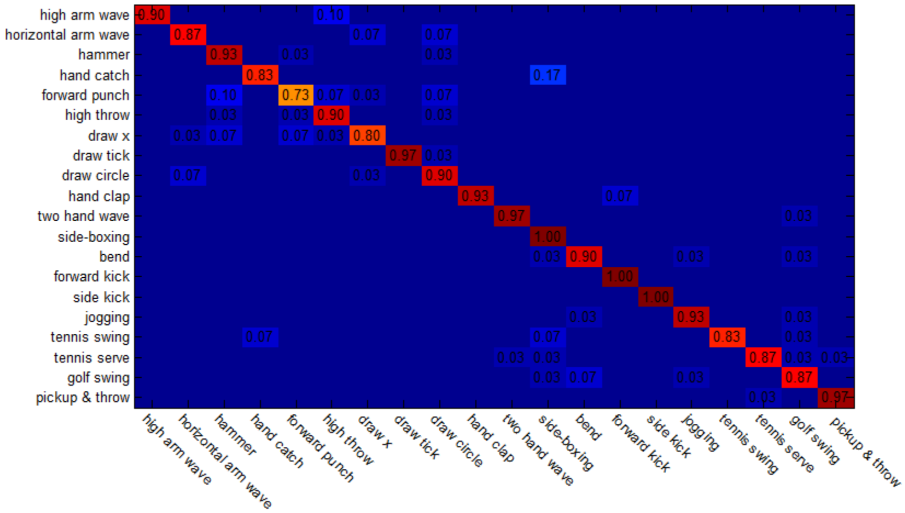
where $k_m(\cdot, \cdot)$ is one of the 10 optional kernel functions, and $d = (d_1, d_2, ..., d_{20})^T$ ($\|d\|_p = 1, p \geq 1$) is the kernel combination vector to be optimized during MKL.

Practically, two types of features are extracted from each frame t: the motion features $m_i[t]$ from 3-D distal limbs segments, and the shape features $s_i[t]$ from spherical harmonics coefficients. For each feature two kernel functions were used: Gaussian function and polynomial function, each with different parameters are linearly combined to classify the action using multiclass-SVM classifier. In addition, the regularization parameter C is fixed to the value 1024 for SimpleMKL algorithm in multiclass-SVM. Within this frame, several experiments were conducted using different number of training samples in order to evaluate the performance of our proposed activity recognition method. The empirical results show that our proposed method is comparable with the state-of-the-art-methods.

6.3.1 MSR-Action 3-D dataset

MSR-Action 3-D dataset [11] is an action dataset of depth map sequences captured by a depth camera (RGBD sensor). This dataset contains different human actions: *high arm wave, horizontal arm wave, hammer, hand catch, forward punch, high throw, draw x, draw tick, draw circle, hand clap, two hand wave, side boxing, bend, forward kick, side kick, jogging, tennis swing, tennis serve, golf swing, pick up and throw*. It includes twenty actions performed by ten subjects. Each action was performed 2 or 3 times by each subject. The size of the depth map is 320 × 240. All the subjects were facing the camera during the performance. The subjects were given a freedom to perform the actions at their own place in front of the camera. Furthermore, most of the actions in this dataset involve the movement of limbs (i.e. arm, leg) in one place, which makes most of the actions highly similar to each other. In fact, this dataset is more challenging compared with the previous datasets such as Weizmann and KTH datasets in human action recognition.

In the first experiment, we employed the end points of distal limb segment to calculate the orientation as a unit vector representing the distal limb segment and the translation with respect to the initial frame. Also, the surface points of the silhouette were employed to calculate the spherical harmonics coefficients. In addition, we trained two third of samples and the other third of the samples were used to test the recognition rate of our proposed method in classifying the performed actions. We repeated this experiment three times, each time we change the training and testing samples, and the results of classifying actions were averaged. Figure 6.4 illustrates the results of classifying actions in form of confusion matrix. From the Figure 6.4, we note that the proposed method achieves an accuracy of 0.907 for all actions together. Also, we observe that the classification errors occur if there are similarity among the actions, such as "forward punch" and "hand catch" or if the occlusion occur among human limbs which makes the skeleton tracker fails as in tennis swing action. Therefore, since the skeleton tracker sometimes fails and because of the high rate of similarity among the actions, we considered the accuracy 0.907 for twenty actions together is a good performance and comparable with other methods.



	high arm wave	horizontal arm wave	hammer	hand catch	forward punch	high throw	draw x	draw tick	draw circle	hand clap	two hand wave	side-boxing	bend	forward kick	side kick	jogging	tennis swing	tennis serve	golf swing	pickup & throw
high arm wave	0.90					0.10														
horizontal arm wave		0.87					0.07		0.07											
hammer			0.93		0.03				0.03											
hand catch				0.83								0.17								
forward punch			0.10		0.73	0.07	0.03		0.07											
high throw			0.03		0.03	0.90			0.03											
draw x		0.03	0.07		0.07	0.03	0.80													
draw tick								0.97	0.03											
draw circle		0.07					0.03		0.90											
hand clap										0.93				0.07						
two hand wave											0.97								0.03	
side-boxing												1.00								
bend												0.03	0.90			0.03			0.03	
forward kick														1.00						
side kick															1.00					
jogging													0.03			0.93			0.03	
tennis swing				0.07								0.07					0.83		0.03	
tennis serve											0.03	0.03						0.87	0.03	0.03
golf swing												0.03	0.07			0.03			0.87	
pickup & throw																		0.03		0.97

Figure 6.4. The confusion matrix for MSR-Action 3-D dataset

In addition, we compare the performance of our proposed method with the state-of-the-art methods which are evaluated on the same dataset (MSR-Action

3-D dataset [11]). The first approach was developed by Xia et al[33], they modeled the histograms of 3-D skeleton joint locations by discrete hidden Markov models in order to train/classify the action. The second approach, Li et al. [11] proposed a Bag-of-3-D-Points model for action recognition.

In order to make a fair comparison, we follow the same experimental settings as [33, 11] by dividing the twenty actions into three subsets, each subset contains eight actions, as listed in Table 6.1. We compare our method with the state-of-the-art methods in two tests. In the first test, two third of the samples were used as training samples and the rest as testing samples. In the second test, half of the subjects were used as training and the rest of the subjects were used as testing. Table 6.2 reports respectively the average recognition rates for those methods.

Table 6.1. The three subsets of actions used in comparison.

Action Set 1 (AS1)	Action Set 2 (AS2)	Action Set 3 (AS3)
Horizontal Wave	High Wave	High Throw
Hammer	Hand Catch	Forward Kick
Forward Punch	Draw X	Side Kick
High Throw	Draw Tick	Jogging
Hand Clap	Draw Circle	Tennis Swing
Bend	Hands Wave	Tennis Serve
Tennis Serve	Forward Kick	Golf Swing
Pickup Throw	Side Boxing	Pickup Throw

As shown in Table 6.2, all methods have a high accuracy in the first test (test-1). The overall recognition rate of our method is comparable with the other methods. In the second test (test-2), the overall accuracies are low for all methods because the number of training samples is low and the similarity among some actions. The significant improvement of our method is likely due to incorporating different types of features in different space as a new way of recognizing human activities.

Table 6.2. Recognition Accuracy Comparison Using MSR-Action3-D dataset.

	Xia et al. CVPR 2012		Li et al. CVPR 2010		Our Method	
Subset	Test-1	Test-2	Test-1	Test-2	Test-1	Test-2
AS1	0.986	0.879	0.934	0.729	**0.932**	**0.743**
AS2	0.979	0.854	0.926	0.719	**0.945**	**0.768**
AS3	0.949	0.634	0.963	0.792	**0.956**	**0.867**
Overall	0.971	0.789	0.941	0.747	**0.944**	**0.797**

6.3.2 MSR-Daily activity 3-D dataset

In another experiment, MSR-Daily Activity 3-D dataset [12], which is captured by a Kinect, was utilized to evaluate the performance of our method for recognizing different human activities. This dataset is created to cover daily activities and human-object interactions in the living room. Thus this dataset is more challenging than the first dataset because of human-object interactions. It contains sixteen different human activity: *drink, eat, read book, call cellphone, write on a paper, use laptop, use vacuum cleaner, cheer up, sit still, toss paper, play game, lay down on sofa, walk, play guitar, stand-up, sit-down*, each subject performs an activity in two different poses: standing pose and sitting on sofa pose. The total number of the samples in each pose is 160 sample, each subject has one sample per one activity. A few sample frames of MSR-Daily Activity 3-D dataset are shown in Figure 6.5. In our experiments, we used the standing pose to evaluate the performance of our proposed method since the segmentation of the dataset is not clean in the sitting on sofa pose and skeleton tracker more noisy. We employed the end points of distal limb segment to calculate the orientation as a unit vector, the 3-D joint positions, and the surface points of the silhouette to calculate the spherical harmonics coefficients.

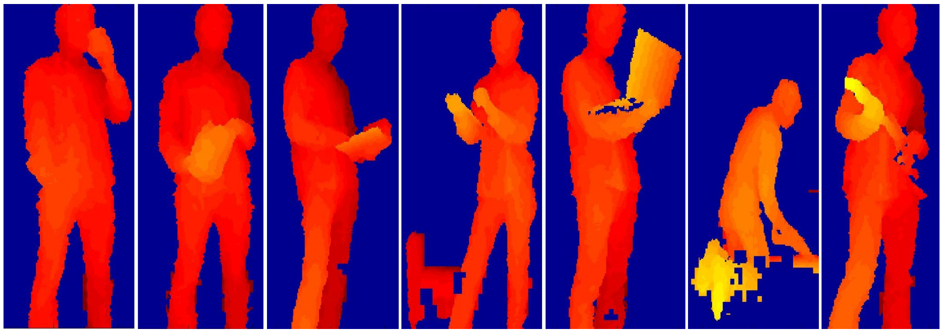


Figure 6.5. A few sample frames of MSR-Daily Activity 3-D dataset; the activities from left to right: drink, eat, read book, play game, use laptop, use vacuum cleaner, play guitar.

Since this dataset has small number of samples for training, Leave-One-Subject-Out (LOSO) test and cross subject test were applied in our experiments to verify the performance of our method. In the LOSO test, one subject is removed from the training set and the other subjects were utilized to train the multiclass-SVM classifier. The excluded subject is used to test the accuracy of our method in classifying the performed activities. The test subject was different from the training ones. This process was repeated for all the subjects (10 subjects), and the results of classifying actions are averaged among ten subjects. In the cross subject test, half of the subjects were used as training data and the other subjects were used as testing data. We repeated this experiment two times, each time we change the training and testing samples, and the results of classifying actions were averaged. The average recognition rate for LOSO test and cross subject test is 0.881, and 0.856 respectively for all activities together. From the Figure 6.6, we can observe that there are a confusion in some similar activities, such as write on a paper and use laptop. In fact, the significant improvement of our method comes in employing the spherical harmonic representation, which describes the interaction between the human body parts and environmental objects.

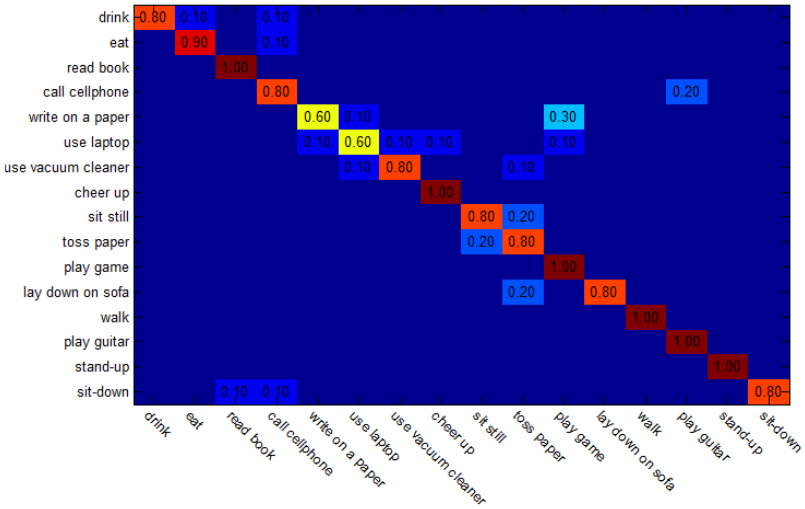


Figure 6.6. The confusion matrix for MSR-Daily Activity 3-D dataset.

Table 6.3 compares the accuracy of the proposed method with the state of art methods on cross-subject test. From Table 6.3, we note that the recognition accuracy of actionlet ensemble model is 0.857, and Fourier Temporal Pyramid method is 0.780. Regarding to our method, which employing the spherical harmonic representation, we obtain a recognition accuracy of 0.856. By considering the difficulties of the dataset and the noisy of 3-D joint positions, we considered the result is reasonably and comparable with the state of art methods.

Table 6.3. Recognition Accuracy Comparison Using MSR-Dialy Activity 3-D dataset on cross subject test.

Method	Accuracy
SVM on Fourier Temporal Pyramid [12]	0.780
Actionlet Ensemble Model [12]	0.857
Proposed Method	0.868

6.3.3 Discussion

In order to evaluate the performance of our proposed method using multiple types of features with a SimpleMKL based multiclass-SVM classifier on human action recognition, three independent SVM classifiers are adopted in our experiments. The first classifier C_1 is C-SVM classifier with a single kernel function and a group of kernel parameters. In this classifier C_1, we used only spherical harmonics coefficients as shape features of human body. The second classifier C_2 is the same as classifier C_1 except the orientation of distal limb segments are used as motion features instead of shape features. The third classifier C_3, which represents our proposed method, is built based on a SimpleMKL based multiclass-SVM classifier. Classifier C_3 is designed with two kernel functions and two types of features including shape features and motion features. This classifier C_3 used to create one kernel weight vector for activity classification.

In our work, we tuned 20 alternative kernel function parameters for Gaussian kernel function and polynomial kernel function and the parameter C is set to

1024 during the training step. However, by fusing shape features and the motion features, our proposed method using a SimpleMKL-based multiclass-SVM can automatically learn the optimal kernel combination weights. This advantage comes from utilizing multiple types of features and different kernel functions, which enhance the performance of the classifier system in the kernel level.

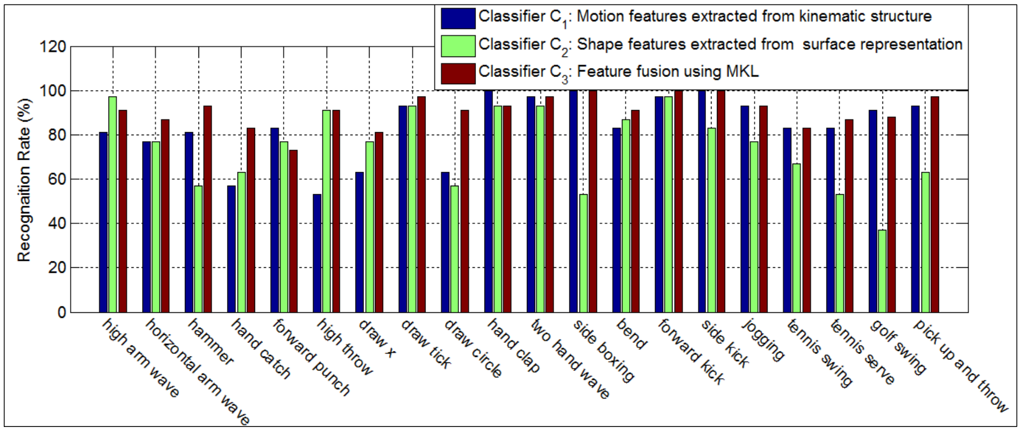


Figure 6.7. Comparison among shape features, motion features and feature fusion using MKL technique.

Figure 6.7 illustrates the calculated confusion matrix for the three classifiers C_1, C_2 and C_3 using MSR-Action 3D dataset [11]. By comparing the results, we can observe that our proposed method using SimpleMKL-based multiclass-SVM can generally boosts the recognition rate of human activity recognition by fusing multiple types of features with different kernel functions and parameters as shown in Figures 6.7. Specifically, the recognition rate for most actions have been increased from 74.1% for classifier C_1, and 83.7% for classifier C_2 to 90.7% using classifier C_3 which represents multi-features with MKL.

Based on the comparison among the experimental results of the three classifiers, we observed the SimpleMKL-based multiclass-SVM increases the recognition rate of human activity recognition, and outperforms single kernel function and single feature type. In addition, the proposed framework can be applied to other applications that have different types of features and kernel functions in classification tasks although our work focused only on one application.

6.4 Summary

This paper presents a novel framework for incorporating different types of features via SimpleMKL algorithm based on multiclass-SVM. We have proposed a novel 3-D feature for human action recognition using sequence of depth maps obtained by Kinect sensor. The proposed features are discriminative enough to classify human activities even with the human object interactions. The proposed method was applied on two public datasets to evaluate the performance of the proposed method. Experimental results have clearly shown the advantage of using MKL technique in combining different types of features.

In addition, our work presents the advantage of using depth maps for human action recognition. Furthermore, we have presented the concept of spherical harmonics representation to extract a 3-D shape features using spherical harmonic decomposition, and the distal limb segments to extract a 3-D motion features.

Chapter 7

Conclusion and Future Research Direction

7.1 Summary and contributions

In this dissertation, we have introduced a new methodology for human action recognition using motion features extracted from kinematic structure, and shape features extracted from surface representation of the human body. The recently released RGB-depth sensor, Kinect, has allowed simultaneous acquisition of RGB image (2-D) and depth map (3-D) with a good resolution 320×240 pixels. Such a sensor allows acquiring multi-modal videos of human movement, hence illustrating human movement using multiple descriptive features. In fact, there are advantages in using multiple types of features for human action recognition, especially if the features are complementary to each other (e.g. kinematic/motion features and shape features). For instance, challenging problems such as inter-class similarity among actions and performance variation, which cannot be resolved easily by using a single type of feature, can be handled by fusing multiple types of features.

To represent the kinematic structure of the human body from RGB images, an approach for detecting/tracking distal limb segments using the kinematic structure is developed. The idea behind using distal limb segments, such as forearms and shins, is to convert the kinematic structure of the human body into discriminative features that can provide sufficient and compact information to recognize human actions. In this manner, a nine segment skeleton model is fitted to the medial axis of the human body to detect the torso and limbs. This nine segment skeleton model is used to create motion features based on the end-points of the distal segments. To evaluate our approach, three action datasets with different resolutions are utilized and the results are compared with state-of-the-art methods.

On the other hand, the surface representation is an important cue that can be helpful in characterizing the structure of the body silhouette. This dissertation presents a new method for representing the human body surface provided by depth map (3-D) using spherical harmonics representation. The advantage of using the spherical harmonics representation is to represent the whole body surface into a finite series of spherical harmonics coefficients. Furthermore, these series can be used to describe the pose of the body using the phase information encoded inside the coefficients. In our approach, surface parameterization is applied to the body surface to achieve one-to-one mapping from the body surface to a unit sphere in order to create spherical function. Then, spherical harmonic expansion is used to represent the spherical function into a set of spherical harmonic coefficients in the frequency domain. These coefficients are utilized as shape features to train/classify human actions using the canonical C-SVM classifier.

However, the kinematic structure and the surface representation of the human body have their own limitations, and we believe that only a multi-modal approach can provide a robust solution for the problem of human action recognition. In particular, we have studied and provided some useful ideas in human action recognition: 1)Motion features extraction using kinematic structure. 2)Shape features extraction using surface representation of the human body. 3) Multi-features fusion at the kernel level using MKL technique. 4) Experiments on 2-D and 3-D human action recognition using different public datasets (Weizmann, KTH, MSR-Action-3D and MSR-Daily Activity-3D datasets).

In this dissertation, we have introduced a framework for human action recognition based on kinematic structure and surface representation of the human body. We summarize the major contributions of this research as follows:

- A novel approach for fitting distal limb segments into nine-segment 2-D skeleton model. This skeleton model is used to provide precise endpoints of the distal segments which are reduced to centroids for efficient recognition. Each limb centroid is described by its angle with respect to the vertical body axis to create an action descriptor vector to represent the position of the torso and four limb segments. The action descriptor is detected and tracked without any manual initialization.

- Reduction of the kinematic structure to the distal limbs only for action recognition, because we believe the distal limb segments (e.g. forearms and shins) provide sufficient and compact information for human movement.

- Based on the kinematic structure only, the motion of the distal limb segments are used as motion features in the spatial domain.

- Performance evaluation of the motion features with respect to the human action classification using two public dataset (Weizmann and KTH datasets).

- A novel 3-D spatio-temporal features based on spherical harmonics representation for human action recognition. In fact, by representing the depth map of the silhouette in the spherical harmonics domain, the spherical harmonics coefficients can describe the whole body shape as a set of shape features.

- Feature fusion operation is adopted to fuse different types of features using MKL based SVM as a novel concept of feature fusion at the kernel level in order to enhance the classification performance.

- We have evaluated the performance of our multi-modal system for human action recognition using two public datasets: MSR-Action 3-D dataset [11] and

MSR-Daily Activity 3-D dataset [12]. Our experimental results show promising direction in incorporating different types of features using the MKL-based SVM technique.

7.2 Future work

Human action recognition is an important area of research that is continuously progressing. Possible future developments, expansions, and improvements of our presented algorithms in this thesis are as follows:

- Clearly, there are some restrictions of the proposed framework for human action recognition. One limitation is the extraction of human silhouette (i.e., back-ground subtraction). In fact, our approach requires the body silhouette to be accurately segmented from the background. Hence, in order to build a robust system, a strong approach for dynamic segmentation independent of environment needs to be developed and be utilized with our framework.
- Another limitation is the viewing direction of the camera which is fixed in our work. In reality, the camera view angle is not fixed and varies depending on the location of the person with respect to the camera. Therefore, improving the developed system to detect/describe human action from different view angles is strongly needed for human action recognition. In particular, constructing 3-D human body shape by using two RGBD sensors or another

technique will enhance our proposed framework to be view-invariant. Therefore, constructing 3-D human body shape requires more research in order to improve view invariant task.

- Extending the multi-modal approach to incorporate depth map information and RGB information to provide more information and build more robust features that can recognize complex activities. In addition, it is an interesting to study these two challenging topics: human-human interactions and human-object interaction in the area of activity recognition.

- Expanding the current multi-modal approach or developing a new technique to extract more than only shape and motion features from the depth maps. For instance, local features such as SIFT, SURF and HOG can be fused in proposed framework in order to enhance the classification performance.

- Extracting and constructing 3-D human body shape from single view angle is one of the most challenging problems that requires more attention and research. Therefore, improving the efficiency of 3-D human body shape for view invariant task makes the developed system more robust and works under pose variations.

- Improving the efficiency of 3-D data to deal with human-human interactions by using Kinect2, which can deal with at most six persons simultaneously. This makes the proposed framework works with multi-agency.

Bibliography

[1] Chen, H.-S., H.-T. Chen, Y.-W. Chen, and S.-Y. Lee (2006) "Human action recognition using star skeleton," in *Proceedings of the 4th ACM international workshop on Video surveillance and sensor networks*, VSSN '06, ACM, New York, NY, USA, pp. 171–178.

[2] Yu, E. and J. Aggarwal (0-0) "Detection of Fence Climbing from Monocular Video," in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, vol. 1, pp. 375–378.

[3] Blank, M., L. Gorelick, E. Shechtman, M. Irani, and R. Basri (Oct.) "Actions as space-time shapes," in *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, vol. 2, pp. 1395–1402 Vol. 2.

[4] Yan, P., S. M. Khan, and M. Shah (2008) "Learning 4d action feature models for arbitrary view action recognition," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, IEEE, pp. 1–7.

[5] Bobick, A. and J. Davis (Mar. 2001) "The recognition of human movement using temporal templates," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **23**(3), pp. 257–267.

[6] Tian, Y., L. Cao, Z. Liu, and Z. Zhang (May 2011) "Chang:2005," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, **42**(3), pp. 313–323.

[7] Li, W., Z. Zhang, and Z. Liu (June) "Action recognition based on a bag of 3D points," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pp. 9–14.

[8] Moeslund, T. B., A. Hilton, and V. Kruger (2006) "A survey of advances in vision-based human motion capture and analysis," *Comput. Vis. Image Underst.*, **104**(2), pp. 90–126.

[9] Poppe, R. (2010) "A survey on vision-based human action recognition," *Image and Vision Computing*, **28**(6), pp. 976 – 990.

[10] for Windows, M. K. (2013), "Microsoft Corporation," `Available online at http://www.microsoft.com/en-us/kinectforwindows/`.

[11] Li, W., Z. Zhang, and Z. Liu (2010) "Action recognition based on a bag of 3d points," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, IEEE, pp. 9–14.

[12] Wang, J., Z. Liu, Y. Wu, and J. Yuan (June) "Mining actionlet ensemble for action recognition with depth cameras," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 1290–1297.

[13] Kläser, A. (2010) *Learning human actions in video*, Ph.D. thesis, PhD thesis, Université de Grenoble.

[14] Niu, F. (2007) *Human activity recognition and pathological gait pattern identification*, Ph.D. thesis.

[15] Weinland, D., R. Ronfard, and E. Boyer (2011) "A survey of vision-based methods for action representation, segmentation and recognition," *Computer Vision and Image Understanding*, **115**(2), pp. 224–241.

[16] Ali, S., A. Basharat, and M. Shah (Oct. 2007) "Chaotic Invariants for Human Action Recognition," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pp. 1–8.

[17] Yu, E. and J. Aggarwal (June) "Human action recognition with extremities as semantic posture representation," in *Computer Vision and Pattern Recognition Workshops, 2009. CVPR Workshops 2009. IEEE Computer Society Conference on*, pp. 1–8.

[18] Chun, S., K. Hong, and K. Jung (2008) "3D Star Skeleton for Fast Human Posture Representation," in *World Academy of Science and Engineering*, vol. 34, pp. 273–282.

[19] Althloothi, S., M. H. Mahoor, and R. M. Voyles (2012) "Fitting distal limb segments for accurate skeletonization in human action recognition," *JAISE*, pp. 107–121.

[20] Fujiyoshi, H. and A. J. Lipton (1998) "Real-time Human Motion Analysis by Image Skeletonization," in *In Proceedings of IEEE WACV98*, pp. 15–21.

[21] Jean, F., R. Bergevin, and A. B. Albu (2005) "Body tracking in human walk from monocular video sequences," in *Computer and Robot Vision, 2005. Proceedings. The 2nd Canadian Conference on*, IEEE, pp. 144–151.

[22] Cohen, I. and H. Li (Oct.) "Inference of human postures by classification of 3D human body shape," in *Analysis and Modeling of Faces and Gestures, 2003. AMFG 2003. IEEE International Workshop on*, pp. 74–81.

[23] Yilmaz, A. and M. Shah (June) "Actions sketch: a novel action representation," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 984–989 vol. 1.

[24] Venkatesha, S. and M. Turk (2010) "Human Activity Recognition using Local Shape Descriptors," in *Pattern Recognition (ICPR), 2010 20th International Conference on*, IEEE, pp. 3704–3707.

[25] Yilmaz, A. and M. Shah (2008) "A differential geometric approach to representing the human actions," *Computer Vision and Image Understanding*, **109**(3), pp. 335–351.

[26] Laptev, I. and T. Lindeberg (Oct.) "Space-time interest points," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pp. 432–439 vol.1.

[27] Efros, A. A., A. C. Berg, G. Mori, and J. Malik (2003) "Recognizing action at a distance," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, IEEE, pp. 726–733.

[28] Scovanner, P., S. Ali, and M. Shah (2007) "A 3-dimensional sift descriptor and its application to action recognition," in *Proceedings of the 15th international conference on Multimedia*, ACM, pp. 357–360.

[29] Wu, G., M. H. Mahoor, S. Althloothi, and R. M. Voyles (2010) "SIFT-motion estimation (SIFT-ME): A new feature for human activity recognition," in *The 2010 International Conference on Image Processing, Computer Vision and Pattern Recognition*.

[30] Chen, M.-y. and A. Hauptmann (2009) "Mosift: Recognizing human actions in surveillance videos," .

[31] Lowe, D. G. (2004) "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, **60**(2), pp. 91–110.

[32] Shotton, J., A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake (June, 2011) "Real-time human pose recognition in parts from single depth images," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 1297–1304.

[33] Xia, L., C.-C. Chen, and J. Aggarwal (June) “View invariant human action recognition using histograms of 3D joints,” in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pp. 20–27.

[34] Yang, X., C. Zhang, and Y. Tian (2012) “Recognizing actions using depth motion maps-based histograms of oriented gradients,” in *Proceedings of the 20th ACM international conference on Multimedia*, MM ’12, ACM, New York, NY, USA, pp. 1057–1060.
URL `http://doi.acm.org/10.1145/2393347.2396382`

[35] Sung, J., C. Ponce, B. Selman, and A. Saxena (May) “Unstructured human activity detection from RGBD images,” in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pp. 842–849.

[36] Yang, X. and Y. Tian (June) “EigenJoints-based action recognition using Nave-Bayes-Nearest-Neighbor,” in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pp. 14–19.

[37] Zhang, H. and L. Parker (Sept.) “4-dimensional local spatio-temporal features for human activity recognition,” in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pp. 2044–2049.

[38] Zhao, Y., Z. Liu, L. Yang, and H. Cheng (Dec.) “Combing RGB and Depth Map Features for human activity recognition,” in *Signal Information Processing Association Annual Summit and Conference (APSIPA ASC), 2012 Asia-Pacific*, pp. 1–4.

[39] Chang, C.-C. and C.-J. Lin (2011) “LIBSVM: a library for support vector machines,” *ACM Transactions on Intelligent Systems and Technology (TIST)*, **2**(3), p. 27.

[40] Ross, A. A., K. Nandakumar, and A. K. Jain (2006) *Handbook of multibiometrics*, vol. 6, Springer.

[41] Mangai, U., S. Samanta, S. Das, P. Chowdhury, et al. (2010) “A survey of decision fusion and feature fusion strategies for pattern classification,” *IETE Technical Review*, **27**(4), p. 293.

[42] Samanta, S. and S. Das (2009) “A Fast Supervised Method of Feature Ranking and Selection for Pattern classification,” *Pattern Recognition and Machine Intelligence*, pp. 80–85.

[43] Sun, Q.-S., S.-G. Zeng, Y. Liu, P.-A. Heng, and D.-S. Xia (2005) "A new method of feature fusion and its application in image recognition," *Pattern Recognition*, **38**(12), pp. 2437–2448.

[44] Yang, J., J.-y. Yang, D. Zhang, and J.-f. Lu (2003) "Feature fusion: parallel strategy vs. serial strategy," *Pattern Recognition*, **36**(6), pp. 1369–1381.

[45] Cortes, C. and V. Vapnik (1995) "Support-vector networks," *Machine learning*, **20**(3), pp. 273–297.

[46] Han, D., L. Bo, and C. Sminchisescu (29 2009-Oct. 2) "Selection and context for action recognition," in *Computer Vision, 2009 IEEE 12th International Conference on*, pp. 1933–1940.

[47] Sun, J., X. Wu, S. Yan, L.-F. Cheong, T.-S. Chua, and J. Li (June) "Hierarchical spatio-temporal context modeling for action recognition," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 2004–2011.

[48] Noguchi, A. and K. Yanai (2010) "A surf-based spatio-temporal feature for feature-fusion-based action recognition," in *Proc. of ECCV Workshop on Human Motion: Understanding Modeling Capture and Animation*.

[49] Senechal, T., V. Rapp, H. Salam, R. Seguier, K. Bailly, and L. Prevost (Aug.) "Facial Action Recognition Combining Heterogeneous Features via Multikernel Learning," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, **42**(4), pp. 993–1005.

[50] Ikizler-Cinbis, N. and S. Sclaroff (2010) "Object, scene and actions: Combining multiple features for human action recognition," *Computer Vision–ECCV 2010*, pp. 494–507.

[51] Gönen, M. and E. Alpaydin (2011) "Multiple kernel learning algorithms," *Journal of Machine Learning Research*, **12**, pp. 2211–2268.

[52] Gonen, M. and E. Alpaydin (2011) "Multiple Kernel Learning Algorithms," *J. Mach. Learn. Res.*, **12**, pp. 2211–2268.
URL `http://dl.acm.org/citation.cfm?id=1953048.2021071`

[53] Chapelle, O. and A. Rakotomamonjy (2008) "Second order optimization of kernel parameters," in *Proc. of the NIPS Workshop on Kernel Learning: Automatic Selection of Optimal Kernels*.

[54] Rakotomamonjy, A., F. Bach, S. Canu, and Y. Grandvalet (2008) "SimpleMKL," *Journal of Machine Learning Research*, **9**, pp. 2491–2521.

[55] Luenberger, D. G. and Y. Ye (2008) *Linear and nonlinear programming*, vol. 116, Springer.

[56] Nocedal, J. and S. J. Wright (1999) *Numerical optimization*, Springer verlag.

[57] Hsu, C.-W. and C.-J. Lin (2002) "A comparison of methods for multiclass support vector machines," *Neural Networks, IEEE Transactions on*, **13**(2), pp. 415–425.

[58] Herda, L., P. Fua, R. Plankers, R. Boulic, and D. Thalmann (2000) "Skeleton-based motion capture for robust reconstruction of human motion," in *Computer Animation 2000. Proceedings*, pp. 77–83.

[59] Stauffer, C. and W. E. L. Grimson (1999) "Adaptive background mixture models for real-time tracking," in *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, vol. 2, pp. –252 Vol. 2.

[60] Jang, B.-K. and R. Chin (1990) "Analysis of thinning algorithms using mathematical morphology," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **12**(6), pp. 541–551.

[61] Remy, E. and E. Thiel (2002) "Medial axis for chamfer distances: computing look-up tables and neighbourhoods in 2D or 3D," *Pattern Recognition Letters*, **23**(6), pp. 649–661.

[62] Phung, S. L., A. Bouzerdoum Sr, and D. Chai Sr (2005) "Skin segmentation using color pixel classification: analysis and comparison," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **27**(1), pp. 148–154.

[63] Yang, M.-H., D. J. Kriegman, and N. Ahuja (2002) "Detecting faces in images: A survey," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **24**(1), pp. 34–58.

[64] Dempster, A. P., N. M. Laird, and D. B. Rubin (1977) "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 1–38.

[65] Schuldt, C., I. Laptev, and B. Caputo (2004) "Recognizing human actions: A local SVM approach," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 3, IEEE, pp. 32–36.

[66] Ning, H., T. Han, D. Walther, M. Liu, and T. Huang (2009) "Hierarchical Space-Time Model Enabling Efficient Search for Human Actions," *Circuits and Systems for Video Technology, IEEE Transactions on*, **19**(6), pp. 808–820.

[67] De Floriani, L. and E. Puppo (1992) "An on-line algorithm for constrained Delaunay triangulation," *CVGIP: Graphical Models and Image Processing*, **54**(4), pp. 290–300.

[68] Brechbühler, C., G. Gerig, and O. Kübler (1995) "Parametrization of closed surfaces for 3-D shape description," *Computer vision and image understanding*, **61**(2), pp. 154–170.

[69] Shen, L. and F. Makedon (2006) "Spherical mapping for processing of 3D closed surfaces," *Image and vision computing*, **24**(7), pp. 743–761.

[70] Shen, L., H. Farid, and M. A. McPeek (2009) "MODELING THREE-DIMENSIONAL MORPHOLOGICAL STRUCTURES USING SPHERICAL HARMONICS," *Evolution*, **63**(4), pp. 1003–1016.

[71] Bjorck, A. (1996) *Numerical methods for least squares problems*, 51, Society for Industrial and Applied Mathematics.

[72] Theodoridis, S. and K. Koutroumbas (2008) *Pattern Recognition*, 4th ed., Academic Press.

[73] Wang, L., H. Zhou, S.-C. Low, and C. Leckie (2009) "Action recognition via multi-feature fusion and Gaussian process classification," in *Applications of Computer Vision (WACV), 2009 Workshop on*, IEEE, pp. 1–6.

[74] Liu, J., J. Yang, Y. Zhang, and X. He (2010) "Action recognition by multiple features and hyper-sphere multi-class svm," in *Pattern Recognition (ICPR), 2010 20th International Conference on*, IEEE, pp. 3744–3747.

[75] Benmokhtar, R. (2012) "Robust human action recognition scheme based on high-level feature fusion," *Multimedia Tools and Applications*, pp. 1–23.

[76] Tran, K., I. A. Kakadiaris, and S. K. Shah (2010) "Fusion of Human Posture Features for Continuous Action Recognition," in *Proceedings of the European Conference on Computer Vision Workshop on Sign, Gesture, and Activity, Crete, Greece.*

[77] Vranic, D. V. (2003) "An improvement of rotation invariant 3D-shape based on functions on concentric spheres," in *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, vol. 3, IEEE, pp. III–757.

[78] Saupe, D. and D. V. Vranić (2001) "3D model retrieval with spherical harmonics and moments," in *Pattern Recognition*, Springer, pp. 392–397.

[79] Shen, L., H. Farid, and M. McPeek (2009) "Modeling 3-dimensional morphological structures using spherical harmonics," *Evolution*, **63**(4), pp. 1003–1016.

[80] Althloothi, S., M. Mahoor, and R. Voyles (2013) "A Robust Method for Rotation Estimation Using Spherical Harmonics Representation," *Image Processing, IEEE Transactions on*, **22**(6), pp. 2306–2316.

[81] Shen, L., S. Kim, and A. J. Saykin (2009) "Fourier method for large-scale surface modeling and registration," *Computers & graphics*, **33**(3), pp. 299–311.

Appendix A

Spherical Harmonics Coefficients

Over the last decade, spherical harmonics coefficients have been applied to several computer vision applications such as 3-D shape descriptors [77] as well as 3-D model retrieval [78], medical image analysis [79] and rotation estimation [80].

According to [79, 81], spherical harmonics coefficients are suitable for shape comparison based on surface representation because it can deal with protrusions and intrusions. Also, it can be used as an abstract features that can characterize the 3-D shape with different resolution depending on spherical harmonics band. In typical cases, a 3-D surface with a few thousand vertices is represented using spherical harmonics coefficients up to a user specified maximum band L_{max}. Thus, spherical harmonics coefficients can be used to represent a 3-D surface as shape features in the frequency domain.

To compute the spherical harmonics coefficients up to a user-desired maximum band for an input spherical function $S_n(\theta_i, \phi_i)$, which is described by a set of n

points, and their spherical function values $i = 1, 2, ..., n$ are described as:

$$S_n(\theta_i, \phi_i) = (S_{x_n}(\theta_i, \phi_i), S_{y_n}(\theta_i, \phi_i), S_{z_n}(\theta_i, \phi_i)) \qquad for 1 \leq i \leq n \tag{A.1}$$

Therefore, we can formulate a linear system based on Equation 5-5 in a form of matrices with the spherical function S, basis functions Y, and spherical harmonics coefficients C as illustrated in equation A.2.

$$\begin{pmatrix} S_1 \\ S_2 \\ S_3 \\ \vdots \\ S_n \end{pmatrix} = \begin{bmatrix} Y_{1,1} & Y_{1,2} & Y_{1,3} & \dots & Y_{1,k} \\ Y_{2,1} & Y_{2,2} & Y_{2,3} & \dots & Y_{2,k} \\ Y_{3,1} & Y_{3,2} & Y_{3,3} & \dots & Y_{3,k} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ Y_{n,1} & Y_{n,2} & Y_{n,3} & \dots & Y_{n,k} \end{bmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_k \end{pmatrix} \tag{A.2}$$

where $k = L_{max}$ is a number of the spherical harmonics coefficients. More specifically, the equation A.2 can be written in more precise way:

$$\begin{pmatrix} S_1(\theta_1, \phi_1) \\ S_2(\theta_2, \phi_2) \\ S_3(\theta_3, \phi_3) \\ \vdots \\ S_n(\theta_n, \phi_n) \end{pmatrix} = \begin{bmatrix} Y_0^0(\theta_1, \phi_1) & Y_1^{-1}(\theta_1, \phi_1) & Y_1^0(\theta_1, \phi_1) & \dots & Y_l^m(\theta_1, \phi_1) \\ Y_0^0(\theta_2, \phi_2) & Y_1^{-1}(\theta_2, \phi_2) & Y_1^0(\theta_2, \phi_2) & \dots & Y_l^m(\theta_2, \phi_2) \\ Y_0^0(\theta_3, \phi_3) & Y_1^{-1}(\theta_3, \phi_3) & Y_1^0(\theta_3, \phi_3) & \dots & Y_l^m(\theta_3, \phi_3) \\ \vdots & \vdots & \vdots & \dots & \vdots \\ Y_0^0(\theta_n, \phi_n) & Y_1^{-1}(\theta_n, \phi_n) & Y_1^0(\theta_n, \phi_n) & \dots & Y_l^m(\theta_n, \phi_n) \end{bmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_k \end{pmatrix} \tag{A.3}$$

From equation A.3 which represents the relation between spherical function and spherical harmonics coefficients ($S = Y \times C$), we can calculate spherical harmonics coefficients.

$$C = Y^{-1}S \tag{A.4}$$

Since the basis function $Y_l^m(\theta_n, \phi_n)$ is an orthonormal basis function for the space of spherical function $S_n(\theta_n, \phi_n)$ defined on the surface of a sphere, the pseudo inverse matrix Y^{-1} of Y can be computed. Therefore, with the following:

- Spherical function S_n which defines the positions (i.e. latitude (θ_i) and longitude (ϕ_i)) of the sampling points n.
- The number of sample points n for the spherical function.
- The highest order of spherical harmonics band k (normally n ¿¿ k).

,

The spherical harmonics coefficients can be easily computed. In other words, the surface points in spherical coordinate system are defined through a set of spherical harmonics coefficients, $(c^m_{lx_i}, c^m_{ly_i}, c^m_{lz_i})$, where l and m denote the order and degree of the spherical harmonics. These coefficients can be calculated up to a user-desired band L_{max} by solving a system of linear equations to describe 3-D object in the frequency domain. The coefficients are determined using standard least squares estimation.

In our work, we used the spherical parametrization to create the spherical function and the spherical harmonics decomposition to calculate the spherical harmonics coefficients which represents a human body surface. In addition, we used the depth map to capture the 3-D surface points of human body up to a reasonable degree of accuracy.

Vita

Salah Ramadan Althloothi was born in Tripoli Province, Libya, on June 6, 1968. He received his elementary education and secondary education at Tripoli School and his high school education at Alsaba High School. In September 1987, he was admitted to the University of Tripoli at Electrical and Computer Engineering Faculty in Tripoli, Libya, from which he was graduated with the B.S. degree in Computer Engineering in May 1992. He continued his graduate studies in University of Putra, Serdang, Malaysia and was awarded M.S. degree in Computer System Engineering in October 2003.

In August 2008, he was admitted to the Graduate School of the University of Denver, where he was granted the degree of Doctor of Philosophy in Electrical and Computer Engineering in August 2013.