Human Motion Analysis for Efficient Action Recognition

Dissertation

zur

Erlangung des Doktorgrades (Dr. rer. nat.)

der

Mathematisch-Naturwissenschaftlichen Fakultät

der

Rheinischen Friedrich-Wilhelms-Universität Bonn

vorgelegt von

Abdalrahman Eweiwi

aus

Hebron, Palestine

Bonn 2015

Angefertigt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Rheinischen Friedrich-Wilhelms-Universität Bonn

1. Gutachter:	Prof. Dr. Christian Bauckhage
2. Gutachter:	Prof. Dr. Jürgen Gall
Tag der Promotion:	14.04.2015
Erscheinungsjahr:	2015

Abstract

Automatic understanding of human actions is at the core of several application domains, such as content-based indexing, human-computer interaction, surveillance, and sports video analysis. The recent advances in digital platforms and the exponential growth of video and image data have brought an urgent quest for intelligent frameworks to automatically analyze human motion and predict their corresponding action based on visual data and sensor signals.

This thesis presents a collection of methods that targets human action recognition using different action modalities. The first method uses the appearance modality and classifies human actions based on heterogeneous global- and local-based features of scene and human-body appearances. The second method harnesses 2D and 3D articulated human poses and analyizes the body motion using a discriminative combination of the parts' velocities, locations, and correlations histograms for action recognition. The third method presents an optimal scheme for combining the probabilistic predictions from different action modalities by solving a constrained quadratic optimization problem.

In addition to the action classification task, we present a study that compares the utility of different pose variants in motion analysis for human action recognition. In particular, we compare the recognition performance when 2D and 3D poses are used. Finally, we demonstrate the efficiency of our pose-based method for action recognition in spotting and segmenting motion gestures in real time from a continuous stream of an input video for the recognition of the Italian sign gesture language.

Acknowledgments

First and foremost, I would like to express my deep gratitude for my supervisor, Prof. Christian Bauckhage, for his warm encouragement, patience, and immense knowledge, which he gladly shared during my PhD studies. His direct supervision and thoughtful guidance helped me during my research and helped this work reach its current form. I want also to express my deep gratitude to my co-advisor, Prof. Juergan Gall, for the insightful discussions, support, and invaluable guidance during my research.

I would like to acknowledge the financial support of the German Research Foundation (DFG), through the project of "automatic activity recognition in large image databases". I should also mention the invaluable academic and technical support from Bonn-Aachen International Center for Information Technology (BIT), Fraunhofer IAIS, and the University of Bonn.

I extend my gratitude to my colleagues and friends, especially, Muhammed Shahzad Cheema, for his helpful discussions and research collaborations during the last 3 years. I am also grateful to the former and current NetMedia group members in Fraunhofer IAIS, especially, Dr. Christian Thurau, for his invaluable contributions during my research. In addition, I do greatly appreciate all the support I have had from my friends and family, especially, Hani Salah and Mahmoud Eweiwi who helped with proofreading of this thesis and gave me many comments that certainly improved the presentation and the structure of this thesis.

Finally, I owe special thanks to my parents Khalid and Hala for their unconditional support and love, to my wife for her patience and support during my studies in Germany, and to my son Kareem and my daughter Leen for being the cutest kids in the world. Having you around helped me overcome all the critical moments during this period.

Contents

List of Figures

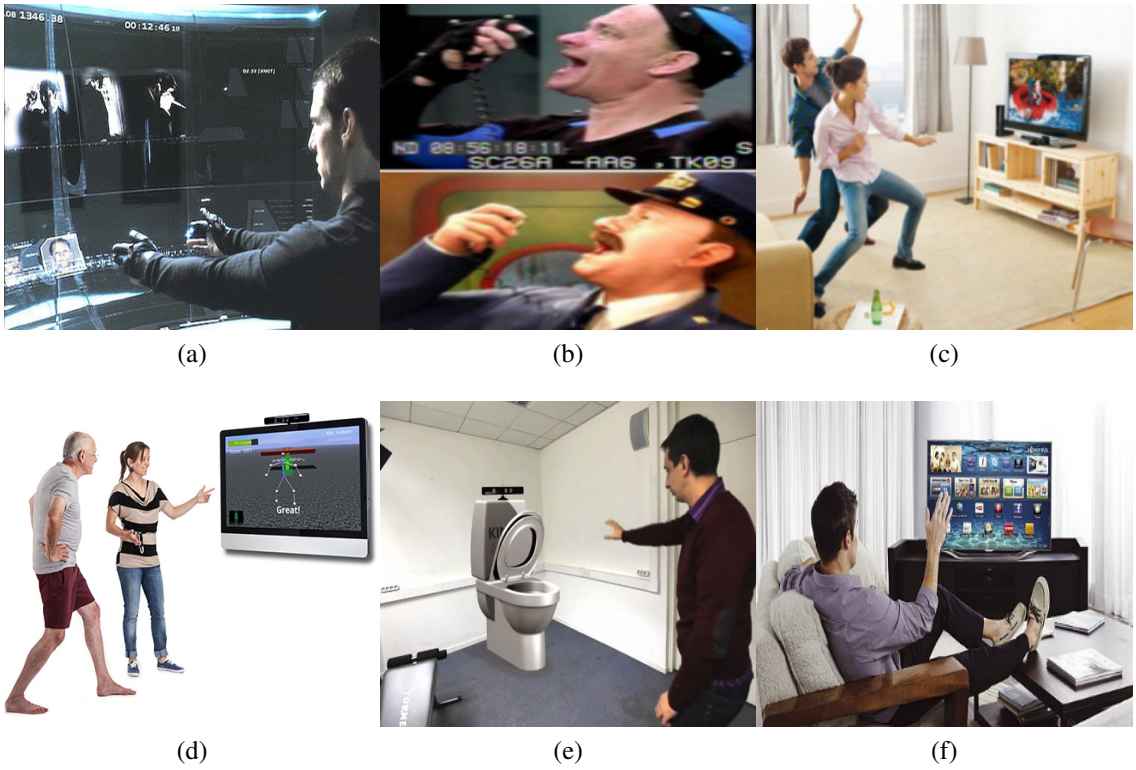
List of Tables

CHAPTER 1

Introduction

Understanding human actions is a remarkable human skill that they can easily perform on a daily basis. Humans constantly try to explain their surrounding environments and interpret behaviors to differentiate between ordinary or alarming activities for development and survival in life. In the machine age, much effort has been devoted towards understanding how the human visual system can effortlessly recognize motion and interpret its meaning to create intelligent machines. The first steps towards understanding human actions can be attributed to the analysis of human motion by *Eadweard Muybridge* (1830-1904), a British photographer, who succeeded in breaking down human actions into distinct, observable body poses. It was not until 1975 when Johansson (1975) provided a psychological understanding of human motion that explains the perception of biological motion by the human visual system. The advent of the computer then brought the quest for intelligent frameworks that can automatically analyze and predict human actions based on visual data and sensor signals. Many use cases appeared to address several challenges ranging from data management and indexing, through automatic-based surveillance systems, to novel approaches for human-computer interaction applications.

With the advances in digital platforms and the exponential growth of video and image data, interest towards automatic human action recognition became intensified as content-based indexing greatly simplifies the manageability of visual data. This advancement has led to greater efficiency in searches. In order to account for the speed of growth in such video data, it is worth noting some recent statistics. For instance, *Facebook* has recently announced



(a) (b) (c)

(d) (e) (f)

Figure 1.1: Action recognition and motion tracking applications: (a) A clip from the *Minority Report* movie which features a "Pre-Crime" police unit that predict crimes before they are committed. The police use a gesture-based interface designed for the film by an MIT media lab team. (b) *Tom Hanks* equipped with motion capture markers to animate the character in the movie: *Polar Express*. (c) *Kinect* game illustration showing motion and action recognition for entertainment applications. (d) An application of human action and gesture recognition for elder care and health control. (e)+(f) Applications of automatic gesture recognition for indoor and public environments.

that they operate one of the largest data warehouses storing more than 300 petabytes of data, the equivalent capacity of as much as 34,245 years of high-definition video. Furthermore, they have reported that more than 500 years worth of *YouTube* videos are watched every day on *Facebook*. *YouTube* has also reported that more than 60 hours of video are uploaded every minute, or one hour of video is uploaded to *YouTube* every second[1]. The same trend goes for *Flickr*, where it has been reported in 2014 that it hosts more than five billion images with an average of 3,000 pictures being uploaded every minute[2]. Unfortunately, to this date, the management and retrieval of such large-scale video or image archives are only possible at the cost of expensive manual annotation.

[1] https://www.youtube.com/yt/press/statistics.html
[2] https://www.flickr.com/photos/franckmichel/6855169886/

The interest in designing automatic human action recognition also goes beyond managing large amounts of data and spans several other fields. Industrial monitoring and surveillance systems using closed-circuit television (*CCTV*) cameras have been largely criticized for their limited role in detecting abnormal and criminal activities. For instance, in the city of London, it has been reported that more than one million *CCTV* cameras have already been installed at a cost of approximately 200 million British pounds. However, despite the high cost, an internal report by the Metropolitan Police of London stated that the installed *CCTV* cameras have not been effective[3]. The report largely cites the manual analysis of video footage by untrained officers to be the major drawback in detecting criminal activities using the *CCTV* cameras.

Exertion game applications, especially for elderly care and health control, may benefit from action and gesture recognition in changing the conventional ways of playing video games for more engaging life experiences (see Figure 1.1). While typical gaming interfaces were based on keyboards, mouses or other haptic-based controllers, human actions and gestures are increasingly used as a direct input. Several approaches have been proposed to advance human-machine interaction. The Microsoft *Kinect* for example, has fostered research in many disciplines of computer vision and human-computer interactions by providing diverse real-time action modalities of depth imagery, voice, RGB, and pose data at affordable price[4].

These examples necessitate automatic action recognition frameworks that can robustly process, analyze, and respond in real-time, if needed, to such challenging scenarios. Unfortunately, despite the extensive research that has been conducted in the field of human action recognition over the past few years, efforts to interpret human actions in images or videos are still in their infancy due to the variation challenges of real-world footage. Variations can be caused by occlusion, viewpoint, scale, background clutter, as well as variation in subject's size, appearance, speed and style of movement. This thesis focuses on methods to overcome these challenges, and proposes several approaches to achieve the construction of a robust, efficient, and low-latency human-action recognition system to support real-time applications.

[3] http://www.telegraph.co.uk/technology/10172298/One-surveillance-camera-for-every-11-people-in-Britain-says-CCTV-survey.html

[4] http://www.xbox.com/en-US/xbox360/accessories/kinect/KinectForXbox360

1.1 Problem Statement

This dissertation focuses on the problem of human-action recognition with respect to different action modalities. While the action representation may be presented in different modalities such as RGB, human pose or body silhouettes, the main goal is to provide an efficient real-time solution of the semantic labels of human actions and investigate the significance of accounting multiple action representations on performance. Therefore, we propose a set of methods that achieves these goals and overcomes some of the challenges in understanding human actions.

To provide an intuitive labeling scheme, action recognition frameworks should follow a natural language convention which uses the typical structure of the sentence: subject, verb, and object. Actions can be defined solely by the verb, for instance "walking" or "running". They can also be used in conjunction with objects such as "playing football" or "drinking coffee". Therefore, it is worth accounting for the difference between action terms that have been interchangeably used to refer to different constructs of human actions. A good taxonomy of human actions by Moeslunda, Hiltonb and Krüger (2006) presents them as a hierarchy that consists of: (i) action primitives and gestures, (ii) actions, and (iii) activities. Action primitives refer to the atomic entities that comprise an action, while actions refer to an ordered sequence of action primitives. Activities, on the other hand, are comprised of a higher level combination of actions that share some temporal relationships between the individual actions. For example, action primitives in the "play tennis" game may be comprised of "run forward", "run backward", "throw ball", and "hit ball", whereas the "tennis serve" action can be described as a set of action primitives that may consist of "throw ball" and "hit ball". The activity of "playing tennis" stands for larger-scale events that usually depend on the context of the environment, objects, or interacting humans.

Thus, in this dissertation, we focus on automatic action recognition of different action constructs using different action representations, and evaluate our proposed approaches based on several benchmarks. In most cases, these benchmarks use appearance, depth, and human pose representation and focus on the lower constructs of the action hierarchy: primitive actions such as "reach" and "release" in the *TUM* dataset (Tenorth, Bandouch and Beetz, 2009) and actions such as "walking" and "running" in the *Web-actions* dataset (Ikizler, Cinbis and Sclaroff, 2009). However, some benchmarks may encompass more complex activities such as "cleaning sofa" in the *MSR-DailyActivity* dataset (Wang et al., 2012a).

1.2 Human Action Representation

The first step in human action recognition is to capture the action's signal using a sensing device or some kind of input representation (modality) that portrays the action. A readily available modality of human actions is their appearance, which is often captured using consumer-like cameras and shared on the Internet. Alternative action representations, such as pose and depth field, also became common due to the recent technological advancements in sensing devices and computer vision algorithms. This section introduces various human action representations that are commonly used for action recognition, describes their advantages and disadvantages, and explains their utility in solving the problem of human action recognition. In this context, we briefly describe the appearance modality (Section 1.2.1), the depth modality (Section 1.2.2), and the pose modality (Section 1.2.3). Note that throughout this thesis, the terms "representation" and "modality" are used interchangeably to refer to the format with which human actions are introduced to the action recognition framework.

1.2.1 Action Appearance

With the advent of digital cameras at the beginning of the twenty-first century, images and videos became a common medium to capture, communicate, and share special moments of our lives. This has led to an exponential growth in volumes of digital media repositories, triggering an urgent necessity of content-based analysis systems to organize and manage such large repositories. In the domain of human action recognition, the application environment plays a decisive role in the design of appearance-based action recognition frameworks. Applications that can influence environmental parameters to a certain degree, such as surveillance, for example, can take certain restricting assumptions for fixed view and limited background clutter. These environments are often referred to as constrained environments. On the other hand, applications that do not have any conditions on the captured appearance modality are often referred to as unconstrained environments. This is the case for most available visual data in TV and cinema movies, sports broadcasts, music videos, or personal footage clips. In such environments, only very few assumptions can be made, such as that humans are fully visible and relatively well displayed in the captured video or image scene. In contrast to constraint environments, unconstrained environments present more challenges as they capture more realistic data and include wide variations in viewpoint, scale, and back-

ground clutter, as well as variations in subjects' appearance, size, and abrupt movement.

1.2.2 Action Depth Field

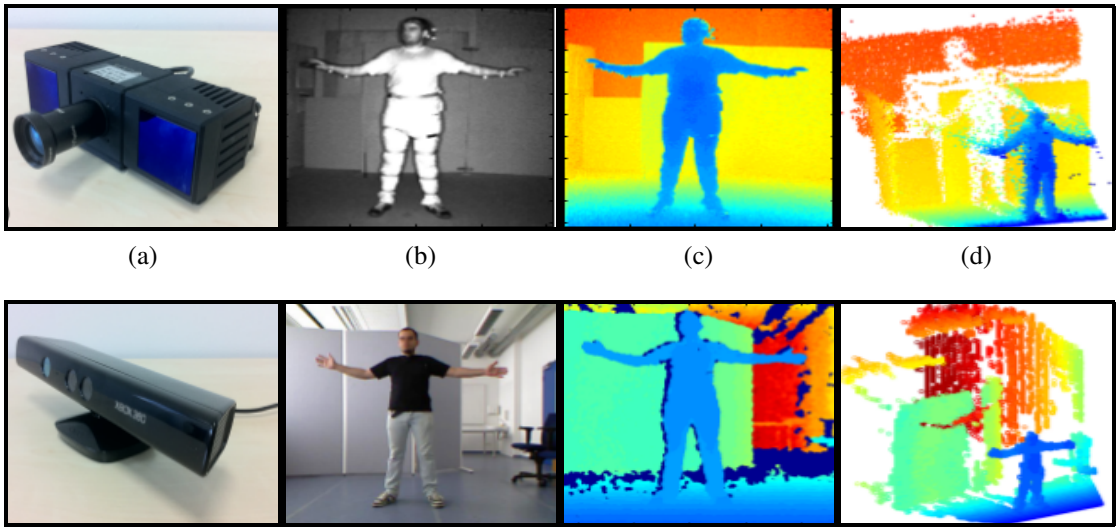
Recent technological advances in sensory data have led to the development of several optical sensors that can acquire three-dimensional scans of a scene in real-time. Previously, obtaining such 3D representation was predominately achieved by carefully setting up a multicamera environment where depth could be reconstructed via triangulations. Nowadays, the newly introduced sensors can easily provide depth information as a measurement of the distance from the camera sensor to the closest object's surface. Two types of optical devices have appeared to obtain depth measurements: Time-of-Flight (*ToF*) and structured light sensors. *ToF* sensors operate similarly to radar where the range image produced is similar to a radar image, except through the use of a light pulse. Cameras, based on the structured light principle, project a known infrared light pattern into the scene and capture the projected pattern using a regular infrared camera. In contrast to *ToF*, structured light sensors, such as the Microsoft *Kinect* sensors are simpler to construct and therefore, comparably less expensive than *ToF* sensors[5]. However, this is subject to change as the technology is rapidly advancing toward designing affordable depth sensors based on both technologies[6].

Despite the wide impact of depth sensors on various computer vision domains, current depth data is still limited, due to several reasons, including: noise, limited maximum-range, artifacts, and data resolution, which is comparably smaller to other optical cameras. For instance, current *ToF* cameras have a resolution of 200×200 pixels, while the *Kinect* sensor captures 640×480 pixels. Moreover, depth sensors can reconstruct only the depth locations that are facing the sensors, i.e. no 3D points are generated at locations where the emitted sensor light can not reach. Therefore, the obtained depth representation is often referred to as 2.5D. Examples of typical depth sensors and their captured depth fields are shown in Figure 1.2.

1.2.3 Action Poses

Poses as an input modality of human motion have been widely used after the pioneering work of Johansson (1975). Johansson (1975) presented a visual interpretation of biological

[5] http://www.xbox.com/en-US/xbox360/accessories/kinect/KinectForXbox360
[6] http://www.intel.com/content/www/us/en/architecture-and-technology/realsense-depth-technologies.html



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 1.2: Typical depth sensors for capturing depth fields: (a) *ToF* sensor, (b) *ToF* amplitude image, (c) Color coded depth image, (d) 3D point cloud, (e) *Kinect* sensor, (f) *Kinect* RGB image, (g) Color coded depth image, (h) 3D point cloud reconstructed from the *Kinect* depth image.

motion which shows that humans are able to interpret their motion solely from the motion of a few moving light displays (MLD) (see Figure 1.3). Various motion-capture approaches use this representation to obtain natural portrayals of human motion using optical tracking systems of markers that are attached to the limbs of the body. While motion-capture systems provide accurate measurements for the movements of the body joints' locations in 3D, they have critical drawbacks that appear when considering realistic unconstrained environments. In such situations, attaching markers to body joints or wearing special suits is impractical for actors who are pursuing some type of daily activity. Recent motion-capture technologies introduced systems that do not demand markers, however, at the cost of careful setup of the environments in terms of visibility and lighting conditions. Typically, these markerless motion-capture systems use multiple regular cameras which are arranged around a common area, and the silhouette of an actor is extracted for each camera view. The extracted silhouettes are employed to reconstruct and estimate the human pose. Unfortunately, these markerless systems demand ideal scene illumination and clear, textureless backgrounds in order to reliably extract body silhouettes.

The environmental restrictions of motion-capture systems motivated automatic-based pose estimation methods to identify and localize different parts of the body from the visual appear-

Figure 1.3: Action examples with moving light displays (MLD) attached to the body's joints

ance of the human body. Comprehensive surveys on recent pose estimation techniques can be found in (Moeslunda, Hiltonb and Krüger, 2006; Sala et al., 2014). Unfortunately, these approaches are still limited in their performance capabilities as they are heavily challenged by the complexities of the high dimension of the search space and the large number of degrees of freedom involved in estimating the body pose. Further complexities arise due to the variation of cluttered backgrounds, body parameters, and illumination changes in real world scenarios. An important milestone towards markerless pose-estimation was achieved after the release of the *Kinect* sensor, where current approaches on pose estimation from depth data obtain a reliable estimation of the human body pose. While providing a good estimation of the human body pose, approaches based on depth data still presume that the entire subject is mostly visible and facing the *Kinect* sensor.

1.3 Contributions

Our research investigates the utility of different action modalities to achieve efficient and reliable human action recognition. It presents novel algorithms and provides extensive empirical evaluation, providing state-of-the-art performance on several action recognition benchmarks. Below we give an overview of our contributions.

We explore the significance of the appearance representation for human action recognition and investigates the benifits of combining different appearance-based features. To this purpose, we present a novel supervised classification framework for action recognition that is based on non-negative matrix factorization (*NMF*). The presented classification framework is a multiclass framework that determines probability estimates of classes for the provided patterns. Therefore, it can efficiently integrate various estimates of different patterns in order to enhance the classification performance. Our research builds on the recent work on

non-negative matrix factorization to multiview learning, where the primary dataset benefits from auxiliary information in obtaining shared and meaningful spaces. For discrimination, we use action labels in a supervised setup as an auxiliary source of information to learn the representative latent set of bases vectors. The evaluation considers an appearance-based approach on two challenging image datasets of human action recognition. In the evaluation, we show how the proposed algorithm achieves competitive classification results. This work was published in (Eweiwi, Cheema and Bauckhage, 2013) and is presented in Chapter 3.

Despite the encouraging performance, the obtained recognition rates using the appearance features only are not ideal for real world application scenarios due the extreme inconsistency of the action appearances. Therefore, we investigate other action representations that can provide better invariance while preserving the distinctive features among semantically different actions. In Chapter 4, we propose a pose-based framework for action recognition that overcomes the varying challenges of appearance-based recognition frameworks. We also show that unlike most previous pose-based approaches, our training and testing time for human action recognition is fast and can meet the demands of real-time applications. Further, the proposed approach achieves state-of-the-art results on several action recognition benchmarks and can work for 2D or 3D pose-based action representations. This work was published in (Eweiwi et al., 2014) and is presented in Chapter 4.

Since most estimation methods for monocular views reconstruct only 2D poses, we compare 2D and 3D pose-based features for human action recognition. We further investigate the significance of reconciling the 2D poses and obtain their corresponding 3D poses for action recognition using a regression scheme. Our study concludes that learning a mapping from 2D poses to 3D poses to obtain view-invariant features can boost the performance significantly. Further, we evaluate the significance of joint orientation features and their role for large scale human action recognition. This work is detailed in Chapter 5 and provides an extension to the work published in (Eweiwi et al., 2014).

As human actions are not usually associated with only the pose representation, but rather on a synergy of representations that captures different action perspectives such as action motion, scale, and scene appearance, we present a novel late-fusion framework that combines several classification results of different action modalities. Our approach is based on formulating and solving a constrained quadratic optimization problem that determines the optimal fusion weights of classifiers based on their operating modality. In contrast to the previously proposed late fusion approaches, our approach puts constraints on the semantics of mixture

coefficients, such that they represent the posterior of every participating classifier for each class. Experiments on a number of established benchmark action datasets show that the presented approach improves on baseline late-fusion approaches and improves on state-of-the-art results. This work is detailed in Chapter 6 and published in (Cheema, Eweiwi and Bauckhage, 2014).

1.4 Related Publications

The following list presents the publications and the contributions made and presented in this dissertation:

[1] A. Eweiwi, S. Cheema, C. Thurau, C. Bauckhage, "Temporal key poses for human action recognition", ICCV-WORKSHOPS, 2011.

[2] A. Eweiwi, S. Cheema, C. Bauckhage, "Discriminative joint non-negative matrix factorization for human action classification", GCPR, 2013.

[3] A. Eweiwi, S. Cheema, C. Bauckhage, "Action Recognition in Still Images by Learning Spatial Interest Regions from Videos", Pattern Recognition Letters 37, 2014.

[4] A. Eweiwi, S. Cheema, C. Bauckhage, J. Gall, "Efficient Pose-based Action Recognition", ACCV, 2014.

[5] S. Cheema, A. Eweiwi, C. Bauckhage, "Action recognition by learning discriminative key poses", ICCV-WORKSHOPS, 2011.

[6] S. Cheema, A. Eweiwi, C. Bauckhage, "Gait Recognition by Learning Distributed Key Poses", ICIP, 2012.

[7] S. Cheema, A. Eweiwi, C. Bauckhage, "Who is Doing What? Simultaneous Recognition of Actions and Actors", ICIP, 2012.

[8] S. Cheema, A. Eweiwi, C. Bauckhage, "Human Activity Recognition by Separating Style and Content", Pattern Recognition Letters 34, 2013.

[9] S. Cheema, A. Eweiwi, C. Bauckhag, "A Stochastic Late Fusion Approch to Human Action Recognition", GCPR, 2014.

CHAPTER 2

Related Work

2.1 Preface

Human action recognition has attracted much attention in the past decade and remains an active research topic in computer vision. The challenge in computer vision, simply stated, is to be able to efficiently and robustly classify human actions. Efficiency denotes the capability of the system in obtaining accurate and fast performance for human action recognition. Robustness refers to the capacity of the system in maintaining its efficiency under unprecedented situations. As noted in (Campbell and Bobick, 1995), the representation of the performed actions often determines the key characteristics of the designed algorithm (i.e. efficiency, robustness, and applicability extents). Ideally, the representation should be invariant towards variations in different actors' styles, views, and backgrounds. Meanwhile, it should preserve the distinctive features among semantically different actions. Following this intuition, we describe the human action recognition challenge from a representation perspective. First, we briefly review current approaches to action recognition, with an emphasis on their used representations (see Figure 2.1). Then, we review the research frontiers on this problem and present exemplar approaches based on the adopted representation. Finally, we elaborate on the advantages and disadvantages of using each representation for delivering efficient and robust human action recognition solutions. From this standpoint, we roughly categorize the proposed methods for human action recognition into two categories based on their corresponding representation:

(a) (b) (c) (d)

Figure 2.1: Different action views of human: (a) appearance, (b) depth field, (c) silhouettes, and (d) pose representations

- **Action recognition using high-level representations:**

  High-level representation of human actions presents the human body pose as a collection of interconnected body parts and joints in a deformable configuration model. These models are offten called body poses or stick-figures. (see Figure 2.1(d)).

- **Action recognition using low-level representations:**

  Low-level representation of human actions presents its visual appearance as an ordered set of pixels of different intensity values for each channel. These channels may correspond to the visual appearance of the action using colored pixels (i.e. RGB), or depth image with pixel intensities that capture the distance of the projected light ray from the real world to the sensor, or body silhouettes where each pixel indicates if it is part of the human body or not.

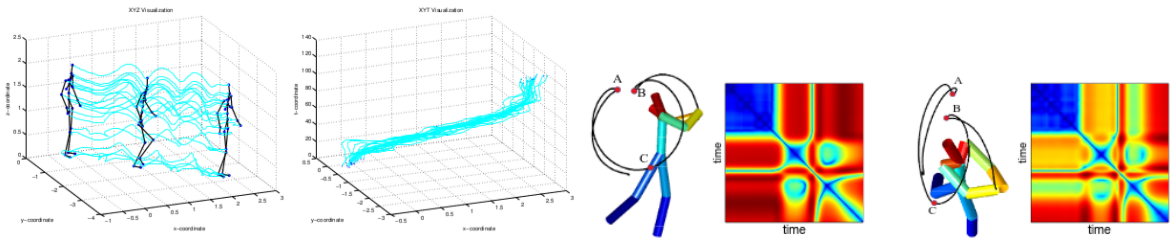
In the following sections, we elaborate further on both approaches and describe some of the recent research made based on these action representations.

2.1.1 Action Recognition Using High-level Representation

Earlier attempts for human action recognition relied on simple human representations called stick-figure models (Johansson, 1975). The stick-figure model is based on a pose structure, where line segments are connected by the body joints to form a hierarchical structure. Obtaining such representation assumes accurate measurements of the body's joints; thus, it often requires special setups and tools. The influence of the psychological studies of human perception of motion (Johansson, 1975) motivated several researchers to account for

this representation in automatic human action recognition. For instance, the approaches in (Campbell and Bobick, 1995; Bissacco et al., 2001; Ali, Basharat and Shah, 2007) use different phase space features extracted from joint trajectories for actions and gaits recognition. Others (Yacoob and Black, 1998; Rao, Yilmaz and Shah, 2002; Junejo et al., 2008) rely on different similarity measures for tracking and matching body joints' trajectories. Parameswaran and Chellappa (2003) propose an invariant feature set for human motion analysis and action recognition using five plenary points of human body joints. Vasilescu and Sethi (2001) pose the human action classification as a model-based object recognition problem using a generalized cylindrical representation called action cylinders. Sheikh, Sheikh and Shah (2005) identify three sources of variability within a performed action and propose to alleviate them through a linear combination model in a joint spatio-temporal space. These approaches, however, demand an expensive and time-consuming setup to operate in order to generate accurate measurements of body joints' locations; therefore, their applicability to real-world environments is limited.

With the recent advances in both depth sensors and automatic human pose estimation algorithms, interest has been rekindled in high-level representations for action and behavior analysis (Ye et al., 2013). Despite their noisy estimations in monocular (Yang and Ramanan, 2011), depth sensors (Shotton et al., 2013), or multiview (Yao, Gall and Gool, 2012) setups, several recent studies (Tran, Kakadiaris and Shah, 2011; Jhuang et al., 2013; Wang, Wang and Yuille, 2013; Wang et al., 2012b; Yao, Gall and Gool, 2012) strongly point to the utility of the pose representation in obtaining superior performance as opposed to low-level features (Section 2.1.2). For example, Tran, Kakadiaris and Shah (2011) utilize polar coordinates of joints in a sparse reconstruction framework to classify human actions in realistic video datasets. Their evaluation clarifies the implication of accurate pose estimation on action recognition and identifies the potentials of current state-of-the-art pose estimation in obtaining excellent recognition performances. Similar observations are reported in (Yao, Gall and Gool, 2012; Jhuang et al., 2013) for larger and more complex datasets. In particular, Jhuang et al. (2013) show that in some scenarios, high-level features extracted using current state-of-the-art pose estimation algorithms (Yang and Ramanan, 2011) outperform best low-level features based on *dense trajectories* (Wang et al., 2011a). These observations motivated researchers to further examine the potentials of high-level features under conventional and newly proposed challenges in videos and depth sensor data. For instance, Wang et al. (2012a) propose learning sets of most distinctive joints through mining. While Zanfir, Leordeanu and



(a) (b)

Figure 2.2: Exemplar features extracted from skeleton-based representation using (a) joints trajectories (Sheikh, Sheikh and Shah, 2005) and (b) trajectory similarity matrix (Junejo et al., 2008)

Sminchisescu (2013) weight poses of actions based on a mutual information criteria, Wang, Wang and Yuille (2013) mine the most occurring temporal and spatial structures.

2.1.2 Action Recognition Using Low-level Representation

Due the practical challenges in obtaining high-level pose representation, research in human action recognition has slowly deviated towards low-level representations of human actions. Two major factors have led towards such a transition. The first is the progress made on low-level features for object detection/recognition. The second is the limited performance of automatic pose estimation algorithms at that time and the high expanses of motion capture setups in obtaining high-level representations. Therefore, alternative low-level action representations were proposed such as body pose silhouettes, action appearances, and depth fields of the action scene. Next, we list exemplar approaches based on each representation and point out the advantages and disadvantages of each representation in action recognition.

Silhouettes-based Approaches

Human body silhouettes were frequently used for human action recognition, especially in the environments where they can be reliably and efficiently captured using background subtraction techniques. A popular work that advocated using this representation was made by Bobick and Davis (2001) where motion and shape cues are combined to create two distinctive action templates called Motion History Images (MHI) and Motion Energy Images (MEI). An extension towards view invariance was proposed in (Weinland, Ronfard and Boyer, 2006) by modeling the human action as 3D template volumes called Motion History Volumes (MHV). Other successful uses of silhouettes were presented in (Thurau and Hlavac, 2007; Thurau et

(a) (b)

Figure 2.3: Exemplar features extracted from silhouettes-based representation as (a) space-time objects (Blank et al., 2005) and (b) action-sketches (Yilmaz and Shah, 2005)

al., 2011; Eweiwi et al., 2011) where they modeled the temporal sequence of human action as a histogram of key poses. These key poses are representative body poses which result from clustering a pool of diverse human body poses. Another histogram-based approach was used by Ikizler, Cinbis and Sclaroff (2009) where they encoded human silhouettes using a histogram of oriented rectangular blocks that span the body pose. Object recognition approaches were also adapted for action recognition through the work of (Yilmaz and Shah, 2005; Gorelick et al., 2007). Their approaches model human actions as spatio-temporal objects that are matched to test samples using a predefined similarity measure. Despite the success made using silhouettes-based approaches, the proposed methods are still bound with the (constrained) environments where silhouettes can be obtained reliably. Moreover, these approaches are still heavily impeded by noise that results from scene-occlusion or inaccurate extraction of the human pose. Therefore, the emphasis of using silhouettes for action recognition has decayed in favor of other representations, such as depth- and appearance-based representations that we describe next.

Appearance-based Approaches

The significant progress made in object and human detection using appearance-based representation (i.e. RGB) promoted several adaptations of motion and visual appearance cues in modeling human actions. Roughly, one can categorize the adopted approaches on modeling human action appearances into:

- **Global template-based models** became popular after the introduction of efficient object and human descriptors, for example, the *Histogram of Oriented Image Gradients*

(*HOG*) descriptor that was presented in (Dalal and Triggs, 2005). In this line, an automatic approach for mining human actions from web images using a variant of the (HOG) descriptors is used in (Ikizler, Cinbis and Sclaroff, 2009). Thurau and Hlavac (2008) apply Non-negative Matrix Factorization (NMF) on the HOG features of pose appearances to learn a set of body-pose primitives. Classification is performed on top of a histogram of pose primitives using Kullback-Leibler (KL) divergence. Efros et al. (2003) use the optical flow fields in constructing global templates of human actions where classification is performed for videos on each frame individually. Motion template features were also used to discriminate among actions, either by encoding the foreground trajectories (Wu, Oreifej and Shah, 2011) or using the optical flow fields (Sadanand and Corso, 2012). In general, global based modeling of human actions works well in delivering the general structure of the human action. But it becomes vulnerable as the variations among actions gets smaller. Moreover, global-based approaches can be severely affected by various impeding factors such as body-parts occlusion, view variations, and background clutter.

- **Part-based models** also present convenient methods for dealing with human action recognition. These methods became popular after their success in many human and object detection challenges (e.g., PASCAL visual object recognition challenge [1]). Felzenszwalb et al. (2010) describe a deformable model for human detection that was used to achieve state-of-the-art performance in action recognition on several benchmarks. Bourdev and Malik (2009) model the human appearance using a set of part-based appearance templates called *poselets* which capture similar pose configurations. Maji, Bourdev and Malik (2011) utilize *poselets* to identify human poses, as well as actions in still images. Sun and Savarese (2011) propose an articulated part-based model for human pose estimation and detection that adopts a hierarchical (coarse-to-fine *poselet*-like) representation. Yang, Wang and Mori (2010) exploit *poselets* as a coarse representation of the human pose and treat them as latent variables for action recognition. The pose-appearance view was also used in (Yao and Fei-Fei, 2012) through a 2.5D representation that considers both pose and appearance information of the body parts. Despite their recent success on different action recognition challenges, it is still questionable if these methods can make use of the favorable statistics of today's large-scale datasets as the construction of suitable *poselets* often requires extensive human intervention and

[1] http://pascallin.ecs.soton.ac.uk/challenges/VOC/

manual labeling in the training phase.

- **Bag-of-Features (*BOF*)** models had been widely used for human action recognition, especially, after the introduction of interest points in video sequences (Laptev, 2005; Willems, Tuytelaars and Gool, 2008). The *BOF* model originated from document and text analysis research in (Salton and McGill, 1986) where documents are represented as a set of orderless words sampled from a language dictionary. The same analogy was used for image and video analysis by constructing a dictionary of visual words to simplify the image or video representation. Briefly, in action recognition, the *BOF* model starts with extracting spatio or spatio-temporal features in the vicinity of random- (Gall et al., 2011), key- (Laptev, 2005; Willems, Tuytelaars and Gool, 2008), dense-points (Sharma, Jurie and Schmid, 2012; Deltaire, Laptev and Sivic, 2010). Then, a dictionary of visual words is constructed using a clustering scheme. The final histogram representation is obtained by encoding and pooling the local features into their corresponding bins. One of the earliest implementations of *BOF* for action recognition in videos was proposed in (Laptev et al., 2008), where spatio-temporal key-points are used to extract local features of human actions. Later experiments adopted a dense-based feature extraction (Reddy and Shah, 2013; Kuehne et al., 2011) which showed better performance in detecting and recognizing human actions.

  Recent work on *BOF* model addresses limitations of the disordered representation of *BOF* model and low-level features (Kovashka and Grauman, 2010). Some methods propose a mid-level representation that conveys more semantics about the actions by encoding spatial and temporal relationships among low-level features. Others (Gilbert, Illingworth and Bowden, 2011; Liu et al., 2012) employ data mining to build high-level compound features from noisy and over-complete sets of low-level spatio-temporal features. Song, Goncalves and Perona (2003) use a triangular lattice of grouped point features to encode spatial layouts. The authors of (Coates and Ng, 2011; Malinowski and Fritz, 2013; Sharma, Jurie and Schmid, 2012) propose weighting schemes of local features while pooling them in a way that regards the classification task at hand. Unfortunately, these approaches provided limited enhancements over their low-level counterparts as they lack semantic meaning, making the interpretation of their mid-level features difficult. Incorporating the semantics behind the visual appearance of the action appears to be a key-factor in obtaining better action models using *BOF*. Therefore, re-

cent approaches (Matikainen, Hebert and Sukthankar, 2009; Wang, Wang and Yuille, 2013; Wang et al., 2011a) use semantic constructs like motion trajectories instead of key-points for sampling local action features. Jhuang et al. (2013) notice that encoding local appearance and motion features in the vicinity of motion trajectories boosts the performance of *BOF* models. While Wang et al. (2013b) show that accurate estimation of action trajectories further enhances the performance of these models.

Despite encouraging results of the appearance-based representation for action recognition on several datasets, many factors greatly impede the progress in this domain. Among these are the heavy variations of the same action appearance across different view points, different subjects, different scales, and even different scenes. Moreover, low-level features of the appearance representation are often limited in their discriminative power for complex and realistic scenarios as they carry limited semantics of the represented actions.

Depth-based Approaches

Much effort has been devoted recently to developing features for action recognition on depth data due to the reliable, affordable, and rich representation depth fields provide for the human actions. Some approaches that use the depth-based representation adopt appearance-based approaches by assuming the depth-field as an intensity image. These approaches used global-based (Yang et al., 2012) or part-based (OhnBar and Trivedi, 2013) templates to capture action depth characteristics. Other recent approaches follow more delicate methods by mining discriminative depth-based occupancy patterns that are randomly distributed over the body's depth field (Wang et al., 2012c) or only around the body's joints (Wang et al., 2012b). Histogram-based approaches were also used in this domain. For example, Li, Zhang and Liu (2012) represent each depth frame as a bag of 3D points on the human silhouette and employ HMM to model the temporal dynamics. Oreifej and Liu (2013) build histogram-based features based on the normals extracted from the 4D spatio-temporal space of the human body (i.e. XYZ+T). Despite the current limitations of maximum captured depth of current depth sensors (e.g., *Kinect* and *Time-of-Flight*), the representation still presents a unique view of human action that is beneficial, especially for indoor applications of human action recognition systems.

2.1.3 Action Recognition: A Multimodal Approach

In the previous sections, we have introduced several action views of high-level (e.g., body poses) and low-level representations (e.g., appearance and depth fields). Noticeably, each view is characterized by several limitations which prohibit obtaining robust action recognition systems. Therefore, the recent trend for action recognition targets fusing several complementary action representations to cope with different action aspects such as motion, scene, pose, and context. Two approaches commonly are used to achieve fusion. The first is early fusion which combines different representations on the feature level. For instance, methods in (Deltaire, Laptev and Sivic, 2010; Rohrbach et al., 2012; Wang et al., 2011b) combine a variety of heterogeneous representations by simply concatenating feature descriptors. This, however, may undermine the discriminative potential of each individual representation for particular classes. To overcome this limitation, Wang et al. (2012a) follow a principled approach to combine a set of mined action features called *actionlets* using Multiple Kernel Learning (MKL) (Bach, Lanckriet and Jordan, 2004), which assigns different linear or non-linear weights to the feature kernels in order to obtain better similarity measures. A recent evaluation Gehler and Nowozin (2009) show that the simple kernel averaging, a much faster method, can achieve similar results as MKL.

The second approach is late fusion, often called classifier-level fusion. This approach has certain key advantages over other fusion schemes. Firstly, late fusion is generally fast and scalable, especially as the trained system grows to adapt new features. In this case, classifier level fusion requires only the retraining of the fusion part in contrast to feature level fusion where the whole system needs to be retrained. Secondly, it abstracts away details of the underlying classifiers, giving the freedom of selecting arbitrary classification models that best suit a given feature. Baseline approaches for classifier level fusion such as the sum-rule or the SVM-rule (Kittler et al., 1998) have been extensively evaluated for several application (Kittler et al., 1998; Xu, Krzyzak and Suen, 1992). These baselines assume that individual classifier outputs are normalized to an estimate of posterior probabilities so that they can be combined homogeneously (Jain, Duin and Mao, 2000). For instance, Eweiwi, Cheema and Bauckhage (2013) combine the estimated confidences rates of different models trained using actions' pose appearances and scene appearances. While Yao et al. (2011a) combine the confidence estimates of models trained on different pose and appearance features. In summary, both late and early fusion of action features for different representations have

shown significant improvements on human action recognition. Therefore, recent approaches focus on the design of better fusion schemes, other than baseline techniques, that adhere the individual advantages of each action representation allowing for better human action recognition systems (Liu et al., 2013a; Ye, Liu and Chang, 2012).

2.2 Human Actions Datasets

The technological advancement in terms of memory, processing speed, and sensory data has opened new application domains and provided appealing tools for analyzing diverse, complex, and large amounts of visual data. As the demand for automatic analysis of visual media became mandatory for wide application domains (e.g. video indexing human-computer interaction, and surveillance), researchers proposed several benchmarks that have had an increasing complexity throughout the last few years. These datasets consider different assumptions of human actions according to the anticipated application domains of the designed algorithms. Among these assumptions are the scale of the recognition problem, its working environment (i.e. constrained or unconstrained), and the representations of human actions (i.e. RGB, RGB-D, silhouettes, motion capture, and skeleton-based representations).

We divide these benchmarks into two categories based on whether they comprise high-level action representation (i.e. human poses depicted by body skeletons) or low-level action representations (i.e. appearance- and depth-based). Table 2.1 lists the action datasets we used in this dissertation and their main characteristics.

2.2.1 Low-level Representation Benchmarks

Low-level representation depicts the visual appearance of human actions as an ordered set of pixels with different intensity values of different channels. These channels may either correspond to the visual appearance of the action depicted by colored pixels (i.e. RGB), or the depth field through pixels whose intensities correspond to the distance of the projected light ray from the real world to the sensor, or silhouettes where each pixel indicates if it corresponds to a human body or not.

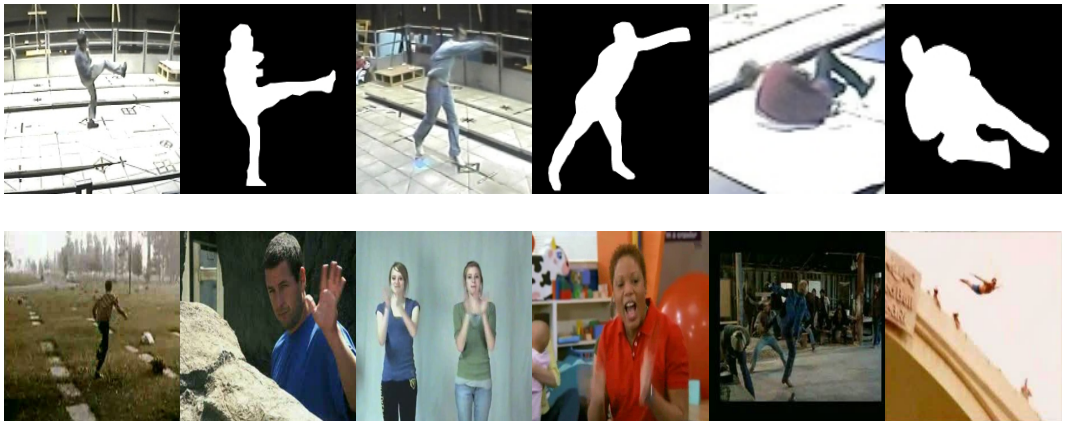


Figure 2.4: Exemplar frames for the MuHVAi human action dataset (first row) with their silhouettes and for the HMDB dataset (second row)

Multi-view Human Action Video (MuHAVi) Dataset

The MuHAVi dataset is a video dataset of two different representations: human silhouettes and RGB data. Silhouette-based representations have been widely used for action recognition in constrained environments. It became popular as it suits particular applications (e.g., surveillance) where reliable human silhouette extraction is possible. The dataset was presented by Singh, Velastin and Ragheb (2010) and considers human actions in a constrained environment. It provides multi-view data of actions of different actors with CCTV-like views (i.e. at an angle and some distance from the observed person). The data consists of 136 samples of 14 primitive actions, performed by two actors, and is observed from two different views. The actions in the data set can be reorganized into eight classes where similar actions constitute a single class. Figure 2.4 shows example frames of this dataset for different human actions in the RGB and the silhouette representations.

Web-actions Dataset

The exponential growth in unconstrained human action videos exposed the potential limitation of silhouette-based representations. These representations are usually intractable for images and unconstrained action videos because of the absence of reliable silhouette extraction methods. As such, several datasets were proposed to recognize human actions based only on their visual appearance (i.e. RGB data). The *Web-actions* dataset is among these datasets that targets action recognition from images gathered for the web. It was presented

Table 2.1: Human action recognition benchmarks and their key characteristics: number of actions, number of samples, type of samples (**I**: Images, **SV**: Segmented videos, and **UV**: Unsegmented videos), year of release, and the available action modalities (**A**: Appearance, **D**: Depth, **P**: Pose, and **S** Silhouettes

Dataset	Actions	Samples	Type	Modality	Year
MuHAVI (Singh, Velastin and Ragheb, 2010)	14	136	**SV**	**S**	2010
Web-Actions (Ikizler, Cinbis and Sclaroff, 2009)	5	2458	**I**	**A**	2009
Willow (Deltaire, Laptev and Sivic, 2010)	7	911	**I**	**A**	2010
HMDB51 (Kuehne et al., 2011)	51	6766	**SV**	**A**	2011
MSR-Action3D [2]	20	576	**SV**	**A+D+P**	2011
MSR-DailyActivity	16	320	**SV**	**A+D+P**	2011
3D-Action-Pairs (Oreifej and Liu, 2013)	12	352	**SV**	**A+D+P**	2013
TUM (Tenorth, Bandouch and Beetz, 2009)	10	20	**UV**	**A+P**	2009
ChaLearn Gestures [3]	20	630	**UV**	**A+D+P+S**	2014

Table 2.2: Example algorithms with their performance on Web-actions dataset

Method	Accuracy(%)	Year
(Ikizler, Cinbis and Sclaroff, 2009)	56.54	2009
(Yang, Wang and Mori, 2010)	61.07	2010
(Eweiwi, Cheema and Bauckhage, 2013)	64.05	2013

by Ikizler, Cinbis and Sclaroff (2009) and contains images downloaded from the Internet using the keywords of human actions. The human body is then extracted using a state-of-the-art human detector and post-processed to align the extracted human bounding boxes with respect to their head position. The resulting dataset consists of five different actions: "dancing", "playing golf", "sitting", "running", and "walking" and contains a total of 2,458 images. Examples from this dataset are shown in Figure 2.5. Pictures in this dataset are characterized by the visibility of human body parts. However, it represents a challenge as the body appearance shows wide pose variations, especially for the "dancing" and "playing golf" actions. Exemplar approaches and their reported results on this dataset are reported in Table 2.2.

Willow Dataset

Advances in social media have revolutionized not only the amount of personal pictures we share on the web, but also provided a diverse view and quality of human visual appear-

Figure 2.5: Examples of different human action images taken from the Web-action (first row) and the Willow (second row) datasets

Table 2.3: Example algorithms with their performance for the Willow dataset

Method	mAP(%)	Year
(Deltaire, Laptev and Sivic, 2010)	62.14	2010
(Eweiwi, Cheema and Bauckhage, 2013)	61.57	2013
(Sharma, Jurie and Schmid, 2012)	65.9	2012
(Delaitre, Sivic and Laptev, 2011)	64.1	2011

ances. The willow dataset[4] proposed by Deltaire, Laptev and Sivic (2010) addresses these challenges by introducing a human action dataset of consumer-like photos that stand for a wide range of variations in view, scene, scale and quality of the visual appearance for people. This dataset consists of 911 images distributed over seven different actions: "interacting with computer", "taking photo", "playing music", "riding bike", "riding horse", "walking", and "running". Some images were taken from the *Pascal 2007 VOC Challenge* and the rest were collected from *Flickr* by querying on keywords such as "running people" or "playing piano". Images that do not clearly depict the action of interest were manually removed. A common observation between the obtained results on the *Web-actions* (see Table 2.2) and the *Willow* datasets (see Table 2.3) is the relatively low performance of the proposed approaches for human action recognition as compared to other datasets that comprise further motion-, depth-, or pose-based representation. This points out to the greater challenges of solving the human action recognition using appearance in images as compared to RGB-videos or other action modalities.

[4] www.di.ens.fr/willow/research/stillactions

Table 2.4: Example algorithms with their performances for the HMDB51 dataset

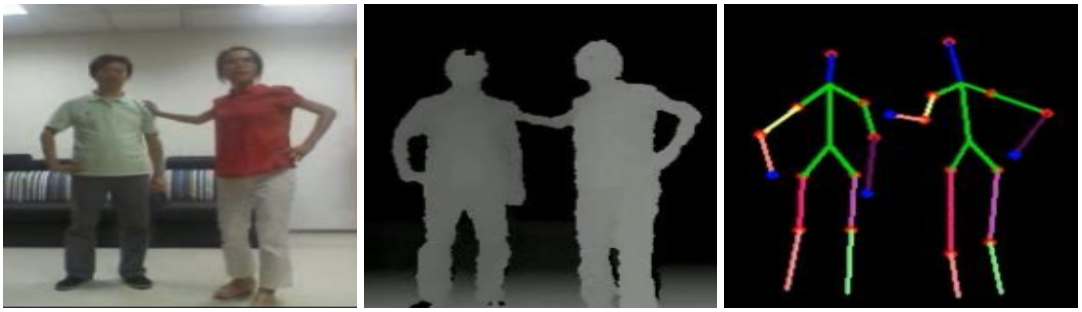
Method	Accuracy (%)	Year
Dense Trajectory (Wang et al., 2013b)	46.6	2013
ActionBank (Sadanand and Corso, 2012)	26.6	2012
MIP (Gross et al., 2012)	29.2	2012
C2 (Kuehne et al., 2011)	23.0	2011
HOG/HOF (Kuehne et al., 2011)	20.0	2011

Large Human Motion Database (HMDB51)

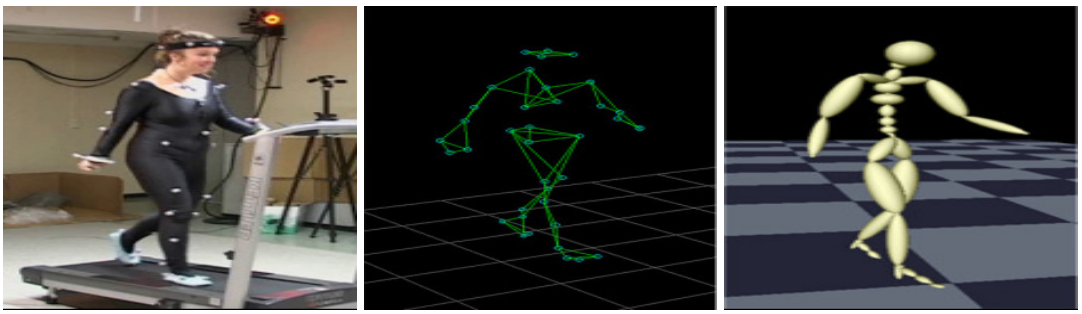
As billions of videos are shared and viewed on the Internet everyday, new frontiers emerged in computer vision to arrange such gigantic growth of media. In contrast to earlier benchmarks for action recognition, *HMDB51* addresses the large scale evolution in media and is considered one of the largest and most challenging benchmarks for action recognition. It comes with 51 distinct action categories each contains at least 101 samples for a total of 6,766 action samples. Each sample clip is validated by at least two human observers and contains additional meta information (i.e. view-point, indicator of camera motion, quality, and the number of actors involved) to provide more flexible experiments for evaluation. Several algorithms were evaluated in this dataset; Table 2.4 shows the state-of-the-art performance achieved in this dataset. Noticeably, the HMDB51 dataset is one of the most challenging benchmarks for action recognition where the best performance of only 46.6% was reported by Wang et al. (2013b) in 2013 using an improved *dense trajectory* features.

2.2.2 High-level Representation Benchmarks

High-level representation of human actions abstracts away most information that is irrelevant to the human body. The representation focuses on modeling the human body and presenting it as a collection of interconnected body parts and joints in a deformable configuration model. Human actions in this depiction can be defined as a collective articulation of the body's joints and parts that uniquely determine the action. This representation is widely adopted in computer graphics, movie production, and animation using motion capture data. Obtaining such representation usually requires special setups that are often time consuming and costly. Figure 2.8a depicts an example of its setup and application for generating high quality computer animations. Recently, with the advent of new sensors (e.g., *Kinect* and *time-of-flight)*, robust algorithms were developed to reliably estimate the human pose in a low-cost monocular



(a) Body poses from depth data



(b) Body poses from mocap data

Figure 2.6: Approaches for capturing abstract human body representations (i.e. body pose) using (a) depth data from the *Kinect* sensor and (b) special motion capture setup

view setup (Figure 2.7b). As a result, researchers presented several challenges that provide not only low-level representations (e.g. RGB-D), but also high-level representations (i.e. body poses) of human actions. In the following sections, we present these datasets and the reported state-of-the-art results on each.

MSR-Action3D Dataset

The *MSR-Action3D* dataset is an action dataset captured with an RGB-D camera and designated for gaming-like interactions. The selected actions reasonably cover the various articulation of arms, legs, torso and their combinations. Additionally, if an action is performed by a single arm or leg, the subjects were advised to use their right arm or leg. The subjects were facing the camera during the performance. The dataset consists of 567 temporally segmented action sequences and contains 20 actions; each performed 2-3 times by 10 different subjects. The actions are: “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-

Table 2.5: Recognition accuracies reported for the *MSR-Action3D* dataset. These methods use different action representations of poses (**P**) and depth fields (**D**).

Method	Modality	Accuracy(%)
(Wang and Wu, 2013)	**D+P**	92.67
(Wang et al., 2012b)	**D+P**	88.2
(Wang et al., 2012c)	**D**	86.5
(Oreifej and Liu, 2013)	**D**	88.36
(Zanfir, Leordeanu and Sminchisescu, 2013)	**P**	91.7
(Wang, Wang and Yuille, 2013)	**P**	90.22
(LXia and Aggarwal, 2013)	**D**	89.3

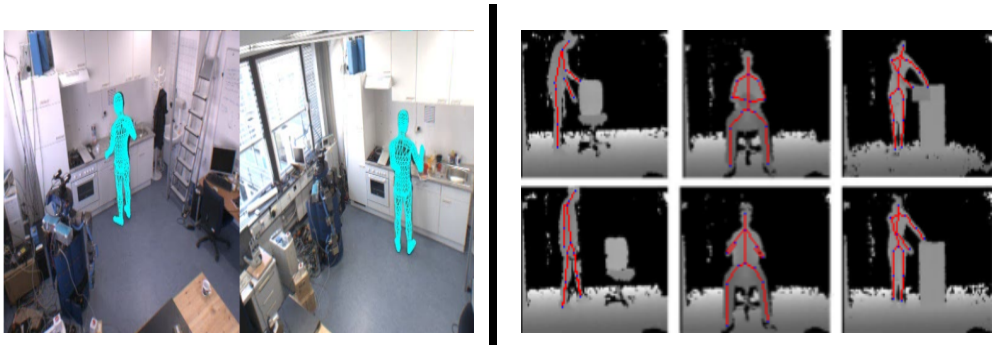


Figure 2.7: Human body-pose exemplar frames for the *TUM* dataset (left) and the *3D Action Pairs* dataset (right).

swing", "tennis-serve", "golf-swing", "pick-up", and "throw". Table 2.5 shows example methods and their reported results using different action representations. Both depth- and pose-based features perform relatively well on this dataset which signify their importance. However, the best results on this dataset (Wang and Wu, 2013) were reported when both depth- and pose-based representations results are combined. This demonstrates the necessity of accounting different action representations in order to achieve better performances in action recognition.

3D Actions Pairs Dataset

This dataset emphasizes particular scenarios where motion and shape cues are highly correlated. It is comprised of six pairs of actions, such that within each pair, the motion and the shape cues are similar, but their temporal correlations vary. Therefore, this dataset is useful to investigate how well the action features capture the prominent cues jointly in the action

Table 2.6: Exemplar recognition accuracy for the *3D Action Pairs*. These methods use different action representations of poses (**P**) and depth fields (**D**)

Method	(Wang and Wu, 2013)	(Wang et al., 2012b)	(Oreifej and Liu, 2013)
Modality	**D+P**	**D**	**D**
Accuracy(%)	97.22	82.22	96.67

Table 2.7: Exemplar recognition accuracies for the *MSR-DailyActivity* dataset. These methods use different action modalities of poses (**P**) and depth fields (**D**).

Method	Modality	Accuracy(%)
(Zanfir, Leordeanu and Sminchisescu, 2013)	**P**	73.8
(Wang et al., 2012b)	**P**	68.0
(Wang et al., 2012b)	**P+D**	85.75
(LXia and Aggarwal, 2013)	**P+D**	88.2

sequence. The action pairs are: "Pick up a box"/ "Put down a chair", "Lift a box"/ "Place a box", "Push a chair"/ "Pull a chair", "Wear a hat"/ "Take off hat", "Put on a backpack"/ "Take off a backpack", and "Stick a poster"/ "Remove a poster". Table 2.6 shows some recently proposed approaches for action recognition in this dataset. Similar to *MSR-Action3D* dataset, the best reported result is obtained when both depth- and pose-based representations are combined.

MSR-DailyActivity Dataset

This dataset was captured by using an RGB-D camera to mimic daily human activities in a living room. There are 10 subjects performing 16 different daily human activities: "drink", "eat", "read book", "call cellphone", "write on a paper", "use laptop", "use vacuum cleaner", "cheer up", "sit still", "toss paper", "play game", "lie down on sofa", "walk", "play guitar", "stand up", "sit down". Each subject performs each activity twice, once in a standing position, and once in a sitting position on a sofa located in the scene. Three data representations are recorded from the human actions: (i) depth maps, (ii) pose joint positions, and (iii) RGB video. The dataset consists in total of 960 files, i.e. 320 video files for each. The provided RGB and depth data representations are recorded independently, so they are not strictly synchronized. The provided body pose representation comprises both real world coordinates (x, y, z) and screen coordinates plus depth (u, v, and depth, where u and v are normalized to be

within [0, 1]). In addition, an integer value is padded at the end to state the confidence value of the captured joint position.

Current state-of-the-art results (see Table 2.7) obtained from this dataset show convenient results when a uni-modal representation of the body pose is used. However, fusing multiple modalities (e.g., body poses and depth fields) may provide an enhanced performance over uni-modal frameworks, especially for actions that share identical properties of pose (e.g., "play game" and "use laptop") or pose and appearance (e.g., "write on a paper" and "read book").

TUM Kitchen Dataset

The TUM kitchen dataset was provided to encourage research in the areas of motion segmentation, markerless human motion capture, and human action recognition. It contains observations for different subjects setting a table in different ways. Some subjects act like a robot, transporting the items one-by-one. Others act more naturally by grasping multiple objects and transporting them together. In general, the *TUM kitchen* dataset focuses on a home-monitoring scenario using a multi-view camera (four cameras). The setup is completely non-intrusive for the human subject, and the recorded sequences are clear of obstructive objects. The dataset provides motion capture data of the subjects using a markerless skeleton tracker. The tracker can reliably track the subjects that interact and manipulate objects even if they were partially occluded by the environment. The dataset also provides different sources of information to better identify the performed actions. These sources include:

1. Video RGB data from 4 different viewing points.
2. Motion capture data of the human poses estimated by a markerless full-body tracker.
3. RFID tag readings from three fixed readers embedded in the environment
4. Magnetic sensors to detect when a door or drawer is opened.
5. Labels of the performed actions in all sequences.

For completeness, we also list the reported results on this dataset in Table 2.8.

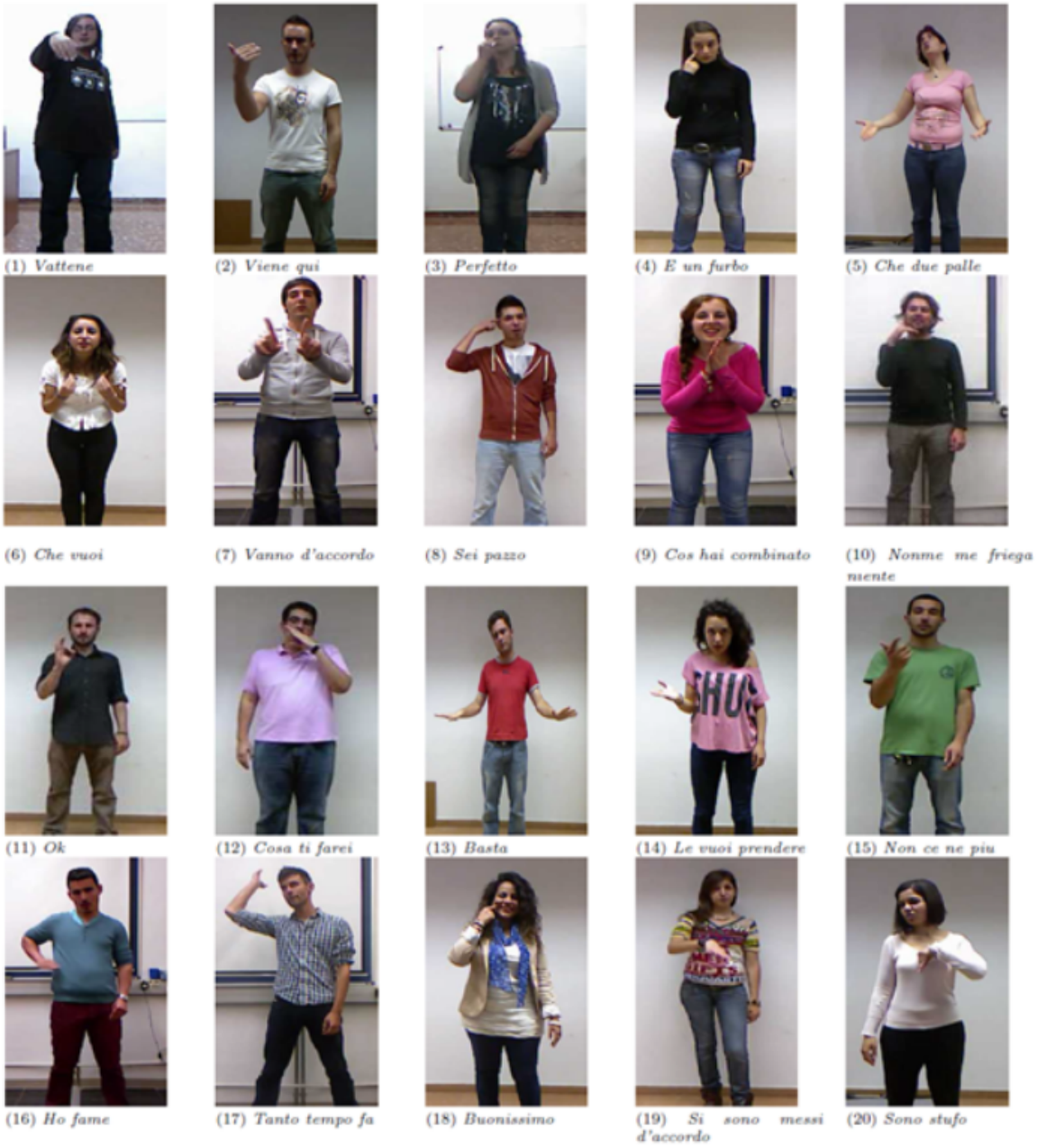


Figure 2.8: *ChaLearn* gesture exemplars for 20 different classes from Italian sign language

Table 2.8: Example algorithms with their performance for the *TUM* dataset

Method	Accuracy (%)	Year
(Yao, Gall and Gool, 2012) using 3D pose	81.0	2012
(Yao, Gall and Gool, 2012) using appearance	71.0	2012

ChaLearn Gesture Dataset

The *ChaLearn* dataset has been recently proposed to address the overwhelming demand of automatic real-time action recognition framework of human gestures from multiple sensory data. It presents several representations of the performed actions including: appearance, depth field, poses (skeletons), and body silhouettes. The objective of this dataset is to design a multi-modal automatic learning algorithm of a set of 20 sign gestures performed by different users, with the aim of performing user-independent, continuous gesture spotting. The dataset is divided into: (i) the development set which comprises more than 7,754 manually labeled gestures, (ii) validation set for cross-validation and model learning and is comprised of 3,362 labeled gestures, and (iii) finally, the evaluation set which is comprised of 2,742 gestures. In total, it stands for almost 14,000 gestures distributed over the 20 classes of Italian sign gesture categories. The 20 gestures' classes in this dataset are: "vattene", "vieniqui", "perfetto", "furbo", "cheduepalle", "chevuoi", "daccordo", "seipazzo", "combinato", "freganiente", "ok", "cosatifarei", "basta", "prendere", "noncenepiu", "fame", "antotempo", "buonissimo", "messidaccordo", and "sonostufo". Each sample sequence corresponds to an actor who randomly selects and performs several gestures among the 20 sign gestures, but he may also perform other undefined movements or gestures. Figure 2.8 shows exemplars of the sign gestures provided for this dataset.

2.3 Summary

This chapter reviewed the human action recognition challenge from a representation perspective. We emphasized the decisive role of the used representation in determining the key characteristics and the applicability extents of the designed algorithm for action recognition. As such, researchers presented several datasets to evaluate the performance attributes of the designed algorithms under the different working environments. These environments can be constrained or controlled, as is often the case for silhouette-based representation, or unconstrained, as is the case for consumer-like videos and images. In the following chapter, we

present the problem of human action recognition using different appearance-based representation of scene and body appearances. Then, we evaluate the proposed methods on several action datasets that were earlier presented in this chapter.

CHAPTER 3

Appearance-based Human Action Recognition

3.1 Preface

Appearance-based representation has recently achieved a considerable interest in human action recognition. It became widely used owing to its success in several challenging vision tasks including pedestrian detection (Dalal and Triggs, 2005; Felzenszwalb et al., 2010), scene analysis(Lazebnik, Schmid and Ponce, 2006), and object recognition (Felzenszwalb et al., 2010). Consequently, research in human action recognition devoted special interest towards analyzing and extracting discriminative appearance-based patterns for action recognition. Often it is the case that the extracted patterns do not correspond only to a particular appearance-view of the human action (i.e. the appearance of the body pose, scene, object or even the body motion), but rather to the synergy of multiple measurements that considers different appearance-based patterns. This chapter follows this incentive by presenting a novel supervised classification approach based on non-negative matrix factorization (*NMF*). The presented classification framework is a multiclass framework that presents probability estimates of classes for the provided patterns. Therefore it can efficiently integrate various estimates of different patterns in order to enhance the classification performance. The proposed framework in this chapter extends the recent work on non-negative matrix factorization to multiview learning, where the primary dataset benefits from auxiliary information for

obtaining shared and meaningful spaces. For discrimination, we use the action labels in a supervised setup as an auxiliary source of information to learn the representative latent set of bases vectors. The evaluation considers two challenging image datasets of human action recognition. In the evaluation, we show how the proposed algorithm achieves competitive classification results. We also demonstrate how the integration of different appearance-based features boosts performance and obtains state-of-the-art in two popular action datasets

3.2 Introduction

Non-negative matrix factorization has been widely used in image analysis and pattern extraction during the last few years. This can be attributed to the convenient interpretation of factorized components and its direct relation to other probabilistic frameworks. Recent research has spanned the applications of *NMF* from retrieval and clustering to other domains including multiview learning. Multiview learning, in the context of retrieval systems for example, profits from auxiliary sources of information in improving retrieval performance on the primary dataset. Such performance gain becomes plausible by estimating meaningful latent structures that explicitly model the co-occurrences between primary and auxiliary data sources. In this sense, one can interpret any supervised classification task as a multiview problem by using the auxiliary category information in extracting discriminative latent structures between different classes.

Despite the venerable tradition of multiview learning in pattern recognition and machine learning, its applications to computer vision and image analysis vision is still limited. The most popular multiview learning approaches, *Canonical correlation analysis* (*CCA*) (Hotelling, 1936) and *partial least squares* (*PLS*) (Wold, 1966), aim at revealing latent components from different modalities that maximally explain the correlation (*CCA*) or covariance (*PLS*) distributions of different views. Donner et al. (2006) harness *CCA* for fast model searching in active appearance models(*AAM*). Kim, Wong and Cipolla (2007) extend *CCA* for tensor analysis of human actions, where similarities among action's videos are measured through joint shared spaces. *PLS* on the other hand has been recently used for modeling the appearance and pose variations in different views (Dondera and Davis, 2011; Haj, Conzalez and Davis, 2012), achieving state-of-the-art results on multiple benchmark datasets. In this chapter, we emphasize multiview *NMF* learning in a supervised setting, where action appearance categories play the role of auxiliary view of the datasets, and classification results

simply approximate the posterior probabilities of target classes.

Multiview learning using *NMF* is receiving increasing interest owing to the convenient and the semantic interpretation of parts for the extracted bases. Akata, Thurau and Bauckhage (2011) learn shared spaces from different views of image datasets through joint non-negative matrix factorization (*JNMF*) for the applications of segmentation and indexing. Caicedo et al. (2012) present an asymmetric algorithm for the construction of shared latent spaces that first derives a semantic representation from the reliable view of the dataset, and then follows by an adaptation over other views. Gupta et al. (2010) argue for limiting the number of shared spaces learned from *JNMF* to cope with the diversity among various data sources. Liu et al. (2013b) propose an *NMF*-based multiview clustering algorithm by searching for a factorization that gives compatible clustering solutions while maintaining meaningful and comparable results across multiple views.

In the domain of human action recognition, it is often that the outcome is not associated with any single view, but rather the synergy of multiple measurements like body pose, appearance, motion, and scene representation. Earlier works on action recognition have generally considered a single view approach for defining the human action (Eweiwi et al., 2011; Ikizler, Cinbis and Sclaroff, 2009; Thurau and Hlavac, 2008; Willems et al., 2009). However, recent studies pointed out the significance of combining multiple views for an accurate modeling of human actions (Yao et al., 2011a). Yao and Fei-Fei (2012) propose coupled features of pose and appearance by learning body part appearance models. In a similar fashion, Maji, Bourdev and Malik (2011); Yang, Wang and Mori (2010) capture local appearances of multiple body parts using *poselets* (Bourdev and Malik, 2009). Others follow a kernelized approach in a Support Vector Machine (SVM) setup to fuse various feature sets obtained from scene and person appearances (Deltaire, Laptev and Sivic, 2010), or motion and scene appearance models (Wang et al., 2011a) to reach to a common consensus of the identity of an action. Despite their good performance on multiple datasets, these approaches have to deal with heterogeneous features of different modalities which are sometimes difficult to combine.

In summary, this chapter presents an approach to classify human actions using their appearances by adapting JNMF for multi-class classification. We also show the efficiency of our approach in integrating various appearance features to enhance the classification accuracy over different benchmark datasets. The rest of this chapter is organized as follows: Section 3.3 reviews the basics of *NMF*, its multiview adaptation (Akata, Thurau and Bauck-

hage, 2011), and our proposed extension for discriminative analysis. Section 3.4 elaborates on the extracted features used for capturing different action modalities. Finally, Section 3.5 presents our evaluation on two benchmark datasets, and compares our approach with other state-of-the-art approaches.

3.3 Non-negative Shared Spaces Learning Via *JNMF*

Our discriminative joint space learning method is formulated using *NMF*. The following sections review the *NMF* algorithm and its expansion to multiview learning using *JNMF*. Section 3.3.3 introduces our proposed approach for using *NMF* for multi-class classification using different appearance features.

3.3.1 Data Factorization Using *NMF*

NMF has been used recently in various image analysis and computer vision fields. It became widely known after Lee and Seung (1999) investigated its proprieties and presented simple algorithms for the factorization. Formally, *NMF* aims to factorize a non-negative data matrix $\boldsymbol{X} \in \mathbb{R}^{N \times M}$ into a product of a basis matrix $\boldsymbol{W} \in \mathbb{R}^{N \times K}$ and its coefficient matrix $\boldsymbol{H} \in \mathbb{R}^{K \times M}$. This factorization can be viewed as a least squares optimization problem, and read as:

$$\begin{aligned} &\min_{\boldsymbol{W},\boldsymbol{H}} \|\boldsymbol{X} - \boldsymbol{W}\boldsymbol{H}\|_F^2 \\ &s.t. \boldsymbol{W}, \boldsymbol{H} \geqslant 0. \end{aligned} \tag{3.1}$$

Both factorized matrices $\boldsymbol{W}, \boldsymbol{H}$ are constrained to be non-negative. In contrast to other factorization techniques such as singular value decomposition (*SVD*) and principal component analysis (*PCA*), the extracted bases $\boldsymbol{W}$ present an intuitive part-based representation for applications where the analyzed matrices consist exclusively of non-negative measurements like color histograms or bag-of-words data representations. These bases also achieve some level of sparsity due to the non-negativity of matrix $\boldsymbol{H}$ as the basis vectors (parts) can only be added and hence participate in a sparse manner to reconstruct the data matrix $\boldsymbol{X}$.

The *NMF* problem as described in Equation 3.1 is a constrained optimization problem which is convex in either $\boldsymbol{W}$ or $\boldsymbol{H}$ but not for both. Therefore, possible solutions are usually not optimal and correspond to local minimal points. The two most popular algorithms for solving the optimization problem of Equation (3.1) are:

Algorithm 1 Multiplicative update algorithm

procedure $NMF(\boldsymbol{X}, K, maxiter)$ ▷ K: number of bases for NMF, $\boldsymbol{X}$: train features, maxiter: maximum number of iteration
  Initialize $\boldsymbol{W}, \boldsymbol{V} \leftarrow$ random matrices.
  for $i = 1$ to $maxiter$ **do**
    $\boldsymbol{H} \leftarrow \boldsymbol{H}. * (X^T X)./(\boldsymbol{W}^t \boldsymbol{W} \boldsymbol{H}$
    $\boldsymbol{W} \leftarrow \boldsymbol{W}. * (X H^T)./(\boldsymbol{W} \boldsymbol{H} \boldsymbol{H}^T$
  end for
end procedure

Algorithm 2 Alternating least squares algorithm

procedure $NMF(\boldsymbol{X}, K, maxiter)$ ▷ K: number of bases for NMF, $\boldsymbol{X}$: train features, maxiter: maximum number of iteration
  Initialize $\boldsymbol{W}, \boldsymbol{V} \leftarrow$ random matrices.
  for $i = 1$ to $maxiter$ **do**
    $\boldsymbol{W}^t \boldsymbol{W} \boldsymbol{H} \leftarrow \boldsymbol{W}^T \boldsymbol{X}$
    Set all negative values to 0 in $\boldsymbol{H}$
    $\boldsymbol{H} \boldsymbol{H}^T \boldsymbol{W}^T \leftarrow \boldsymbol{H} \boldsymbol{V}^T$
    Set all negative values to 0 in $\boldsymbol{W}$
  end for
end procedure

1. **Multiplicative update algorithm (Lee and Seung, 1999)**: The multiplicative update algorithm is the most popular approach for solving the *NMF* optimization problem and is known for its simplicity. However, it often yields suboptimal solutions and requires many iterations to reach convergence. The multiplicative update rule is described in Algorithm 1.

2. **Alternating least squares algorithm (Paatero and Tapper, 1994)**: The alternating least squares algorithm is another approach for extracting positive bases $\boldsymbol{W}$ and coefficient matrix $\boldsymbol{H}$, where an alternating least square optimization is performed with non-negativity constraints between $\boldsymbol{W}$ and $\boldsymbol{H}$. Unfortunately, solving the least square in this problem with the non-negativity constraints significantly increases the cost of the solution. Therefore, researchers settle for the speed offered by simply projecting the extracted $\boldsymbol{W}$ and $\boldsymbol{H}$ back to the non-negative orthant. Detailed description of the work flow of this algorithm is provided in Algorithm 2.

3.3.2 JNMF for Multiview Learning

Recent studies presented several adaptation techniques of the NMF algorithm to multiview learning (Liu et al., 2013a; Caicedo et al., 2012; Gupta et al., 2010; Akata, Thurau and Bauckhage, 2011). Our work is directly motivated by their efforts along with the study of Barker and Rayens (2003) that explains the statistical discrimination capabilities of traditional multiview learning algorithms like canonical correlation analysis (CCA) and partial least squares (PLS). We follow the adaptation by Akata, Thurau and Bauckhage (2011) in learning fully shared spaces among primary and auxiliary representations of the dataset. Formally, they assume different modalities of a given dataset of M samples captured by matrices $\boldsymbol{X} \in \mathbb{R}^{N \times M}$ and $\boldsymbol{Y} \in \mathbb{R}^{L \times M}$. The basic idea of their algorithm is to find K suitable basis vectors $\boldsymbol{W} \in \mathbb{R}^{N \times K}$ and $\boldsymbol{V} \in \mathbb{R}^{M \times K}$ for both modalities that are coupled implicitly via a common coefficient matrix $\boldsymbol{H}$. In other words, the algorithm aims at finding two low rank approximations such that:

$$\boldsymbol{X} = \boldsymbol{W}\boldsymbol{H} \quad \text{and} \quad \boldsymbol{Y} = \boldsymbol{V}\boldsymbol{H} \tag{3.2}$$

The proposed solution can be formulated as a convex combination of two constrained least squares problems

$$\begin{aligned} &\min_{\boldsymbol{W},\boldsymbol{H}} (1-\alpha)\|\boldsymbol{X} - \boldsymbol{W}\boldsymbol{H}\|_F^2 + (\alpha)\|\boldsymbol{Y} - \boldsymbol{V}\boldsymbol{H}\|_F^2 \\ &s.t \quad \boldsymbol{V}, \boldsymbol{W}, \boldsymbol{H} \geqslant 0. \end{aligned} \tag{3.3}$$

where $\alpha \in [0, 1]$ controls the residual error penalty on each factorized view. This optimization objective can be solved using similar rules presented by Lee and Seung (1999), with a small modification to fit the multiview setup. The multiplicative update rules for bases of both views $\boldsymbol{W}, \boldsymbol{V}$ are

$$\begin{aligned} \boldsymbol{W} &= \boldsymbol{W} \odot \frac{\boldsymbol{X}\boldsymbol{H}^{\boldsymbol{T}}}{\boldsymbol{W}\boldsymbol{H}\boldsymbol{H}^{\boldsymbol{t}}} \quad \text{and} \\ \boldsymbol{V} &= \boldsymbol{V} \odot \frac{\boldsymbol{Y}\boldsymbol{H}^{\boldsymbol{T}}}{\boldsymbol{V}\boldsymbol{H}\boldsymbol{H}^{\boldsymbol{t}}} \end{aligned} \tag{3.4}$$

while the update rule for the shared coefficients matrix $\boldsymbol{H}$ among different views factoriz-

ations is

$$\boldsymbol{H} = \boldsymbol{H} \odot \frac{(1-\alpha)\boldsymbol{W}^{\boldsymbol{T}}\boldsymbol{X} + (\alpha)\boldsymbol{V}^{\boldsymbol{T}}\boldsymbol{Y}}{((1-\alpha)\boldsymbol{W}^{\boldsymbol{T}}\boldsymbol{W} + \alpha\boldsymbol{V}^{\boldsymbol{T}}\boldsymbol{V})\boldsymbol{H}} \tag{3.5}$$

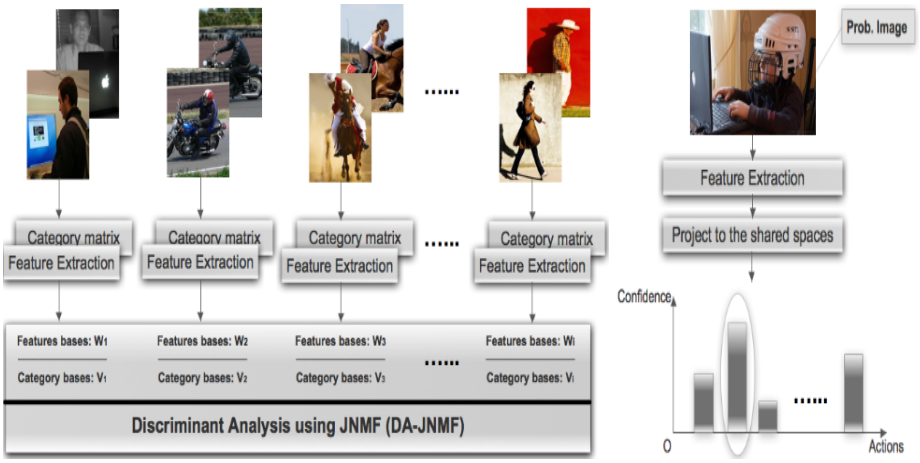
3.3.3 Discriminative Analysis Using JNMF (DA-JNMF)

Direct adaptation of JNMF for statistical discrimination has been investigated in this study, and compared with the proposed approach. In this setup, we assume an annotated feature set for M samples of G different categories, we encode the auxiliary information of group membership of feature matrix $\boldsymbol{X} \in \mathbb{R}^{M \times N}$ using a dummy matrix $\boldsymbol{Y} \in \mathbb{R}^{M \times G}$ as in (Barker and Rayens, 2003) as

$$\begin{bmatrix} \mathbf{1}_{\boldsymbol{m_1}} & 0_{m_1} & \dots & 0_{m_1} \\ 0_{m_2} & \mathbf{1}_{\boldsymbol{m_2}} & \dots & 0_{m_2} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{m_g} & 0_{m_g} & \dots & \mathbf{1}_{\boldsymbol{m_g}} \end{bmatrix}$$

where m_g denotes the number of features of class g. One major drawback of such adaptation is related to the optimization problem itself of Equation 3.3, which aims at minimizing the weighted difference of the Frobenius norm simultaneously for all categories. This adaptation often leads to a quick descent into local minima for both $\boldsymbol{W}$ and $\boldsymbol{V}$. Therefore, the extracted bases fail to capture the discriminative latent space of the training dataset. To remedy this limitation, we suggest proceeding in an incremental fashion where the joint factorization is performed individually for each class. We hypothesize that such a technique results in more discriminative latent structures, and it consequently provides better models for classification. Our empirical results, detailed in Section 3.5, validate this observation over multiple benchmark datasets.

Earlier studies (Ding, Li and Peng, 2008) revealed that by estimating $\boldsymbol{W}\boldsymbol{D}^{-\mathbf{1}}$ or alternatively $\boldsymbol{D}\boldsymbol{H}$ where $\boldsymbol{D} \in \mathbb{R}^{K,K}$ is a diagonal matrix defined as $\boldsymbol{D}_{\boldsymbol{k},\boldsymbol{k}} = \sum_i \boldsymbol{W}_{\boldsymbol{i},\boldsymbol{k}}$ promotes all formal properties of a conditional probability matrix where each column of $\boldsymbol{H}$ defines to which degree feature i is associated for the basis k. Given this fact, we normalize all extracted bases from both views using the diagonal matrix $\boldsymbol{D}$. Empirically, we observed that normalizing the extracted bases led to a slight enhancement on the performance of our algorithm; results are further detailed in Section 3.5.



(a) (b)

Figure 3.1: General diagram of DA-JNMF classification for (a) training on ground truth data and (b) testing on new samples

Algorithm 3 DA-JNMF algorithm

procedure DA-JNMF(X, Y, X_t, k, G) ▷ k: number of bases for NMF, G: number of classes
  Initialize W_G, V_G to empty matrices.
  for $g = 1$ to G **do**
    $W_g, V_g \leftarrow$ solve optimization of (Equation 3.3) using (Equation 3.4) and (Equation 5.2)
    $W_g \leftarrow W_g D_g^{-1}$
    $V_g \leftarrow V_g D_g^{-1}$
    $W_G \leftarrow [W_G|W_g]$
    $V_G \leftarrow [V_G|V_g]$
  end for
  $H_t \leftarrow$ solve for H_t in $X_t = W_G H_t$
  $Y_t \leftarrow V_G H_t$
  return Y_t
end procedure

3.4 Action Appearance Features

To evaluate our proposed algorithm on the problem of human actions, we capture various representations of body-pose and scene appearance. Observing that these multiple features often provide complementary information, it appears natural to integrate them for better performance rather than relying on a single feature representation. As our algorithm provides an approximation of the posterior probability for an action given its features, it would be suitable to integrate those multiple approximations from different features to gain a better performance over each. To this purpose, we utilize different action features that describe the action pose appearance using the Histogram of Oriented Gradients (*HOG*) descriptors and local based feature that describes the scene using Bag-Of-Features (*BOF*) model.

3.4.1 HOG Feature Templates

The *HOG* feature is a rigid template descriptor that count the occurrences of gradient orientation of a set of ordered local image regions. It was proposed by Dalal and Triggs (2005) and has shown to be efficient in capturing the shape and pose appearance in several computer vision challenges including pedestrian detection (Dalal and Triggs, 2005), and human action recognition (Thurau and Hlavac, 2008). The descriptor divides the image region into a small connected regions called *cells*. From each *cell*, the gradient orientation is quantized, aggregated, and contrast-normalized within a larger image region called *block*. As a result, the descriptor provides better invariance against illumination and shadowing. The combination of *block* histograms generates the final *HOG* descriptor. In our experiments, we define the *cell* region to be of size 8×8 pixels, while the *block* region to be of size 2×2 *cells* with an overlap of one among the *blocks*.

3.4.2 Local Action Features

Local features of human action are often used to capture appearance and context features as they are invariant to certain transformation such as translation and scaling. The extraction of local image features constitutes of two main steps:

1. The detection of sampling points using a dense sampling approach (Wang et al., 2011a), or a random-based approach (Tuytelaars, 2010) or based on a measure of their interest (Mikolajczyk and Schmid, 2002; Lowe, 2004). Our experiments follows the recent

convention of using dense sampling grid for defining interest points (Tuytelaars, 2010) as it has shown better recognition performance in several object and action recognition applications (Deltaire, Laptev and Sivic, 2010; Yao et al., 2011b; Yao and Fei, 2010).

2. The extraction of local image features by describing the surrounding regions of the sampled points. The description of these regions can be as simple as an intensity histogram. However, more complex local feature descriptors are often preferred because they account for some degree of invariance against illumination change or geometric distortions. In our experiments, we use the *SIFT* descriptor which computes eight orientation directions over a 4×4 grid which produces a 128-dimensional feature vector. To offer some level of invariance against geometric distortion and noise, Lowe (2004) used Gaussian window function that gives more weight to the gradients computed near the center of the local region. Also, to provide some robustness against illumination changes, the *SIFT* descriptor is normalized to one. After obtaining a set of local action features, the next step is to cluster them using *Kmeans* and obtain a visual codebook. This codebook is used for coding local image features to a different representation by (non) linear operation. The codes are *pooled* afterwords using a pooling operator to obtain the final *BOF* representation. Our implementation uses the recent encoding scheme of *Locality Linear Coding* (*LLC*) proposed in (Wang et al., 2010) and pools the resulting local features codes using a maximum pooling operator. Due to the orderless nature of the *BOF* model, we use a *Spatial Pyramid* binning scheme (Lazebnik, Schmid and Ponce, 2006) over three levels of 1×1, 2×2, and 4×4 to account the spatial distribution of local features while preserving the invariance of the representation against translation and scaling.

Both descriptors were used with our *DA-JNMF* Algorithm 3. The results were later fused by using the sum rule (Kittler et al., 1998) of both trained model outputs and selecting the class that had the maximum confidence as the action of the queried image. The final class Q of an image is obtained as:

$$Q = \arg\max \frac{1}{|v|} \sum_{i}^{|v|} Y_i \tag{3.6}$$

where $|v|$ denotes the number of modalities used to represent an action.

3.5 Evaluation

We evaluate the proposed classification scheme on two benchmark datasets for human action recognition. This section describes the experimental setup, and compares our technique with current state-of-the-art approaches used for human action recognition.

3.5.1 Datasets

We selected two diverse and challenging datasets of human action images to evaluate our proposed approach. Our goal is not limited to show the competitive performance of our classification scheme in terms of accuracy and performance, but also to investigate the merits of integrating multiple views of action images for the purpose of action recognition. The first dataset is the *Willow* dataset[1] (Deltaire, Laptev and Sivic, 2010) (see Section 2.2.1). This dataset comprises of seven different actions "interacting with computer", "taking photo", "playing music", "riding bike", "riding horse", "walking", and "running". The total number of images is 968 split into training, testing, and validation sets. The dataset targets human action recognition in normal consumers photos obtained from *Flickr*. It stands for a wide variation in terms of the human pose, views, and scenes. The second dataset is the *Web-actions* dataset[2] (see Section 2.2.1) is presented by Ikizler, Cinbis and Sclaroff (2009). It contains a total of 2,458 images downloaded from the Internet. We operated on the processed version of the dataset with cropped and aligned human body with respect to head position. The dataset consists of five different actions: "dancing", "playing golf", "sitting", "running" and "walking". We randomly split it into $\frac{1}{3}$ for training and the rest for testing. Pictures in this dataset are characterized by a better visibility of human body parts, but still represent a challenge as they show wide pose variations. Examples from both datasets are shown in Figure 3.2

3.5.2 Results

We followed the experimental procedure proposed by Deltaire, Laptev and Sivic (2010) for the *Willow* dataset. We considered pose appearance captured in terms of human bounding boxes using a three level spatial pyramid with LLC encoding (F.SPM), and the scene view

[1] www.di.ens.fr/willow/research/stillactions
[2] http://cs-people.bu.edu/ncinbis/actionsweb/dataset_release

(a) working on computer (b) taking photo (c) taking photo (d) playing music (e) playing music (f) riding motorbike

(g) walking (h) sitting (i) playing golf (j) dancing (k) walking (l) playing golf

Figure 3.2: Examples of human action images from the Willow action dataset (first row), and the *Web-actions* dataset (second row)

using the original images with the same feature (B.SPM). In both cases, images were resized to a maximum size of 300 pixels before extracting the features. For the *Web-actions* dataset (Ikizler, Cinbis and Sclaroff, 2009), we captured human pose appearance using the *HOG* descriptor while the scene is represented using a three level spatial pyramid. Finally both results were integrated as we have mentioned in Section 3.4. Figure 3.3 depicts the classification accuracy of each action view on both datasets using the proposed approach.

As discussed above, our classification model needs only to specify the number of latent bases K used in the *DA-JNMF* algorithm. We observed a small variation in classification accuracy when K varies between 100 and 400. Setting the parameter K beyond these values results in a worse performance in terms of overall accuracy (if $K < 100$), or in terms of training time (if $K > 400$). Figure 3.3 (a) and (b) show the effect of varying this parameter on overall classification results obtained from both views on both dataset using both *JNMF* and *DA-JNMF*.

Table 3.1 compares our results with state-of-the-art on both datasets. Note that our results on the Web-action dataset significantly outperform the baseline result (Ikizler, Cinbis and Sclaroff, 2009) and state-of-the-art (Yang, Wang and Mori, 2010) as both features capture complementary action proprieties of body pose appearance using the *HOG* descriptor, and

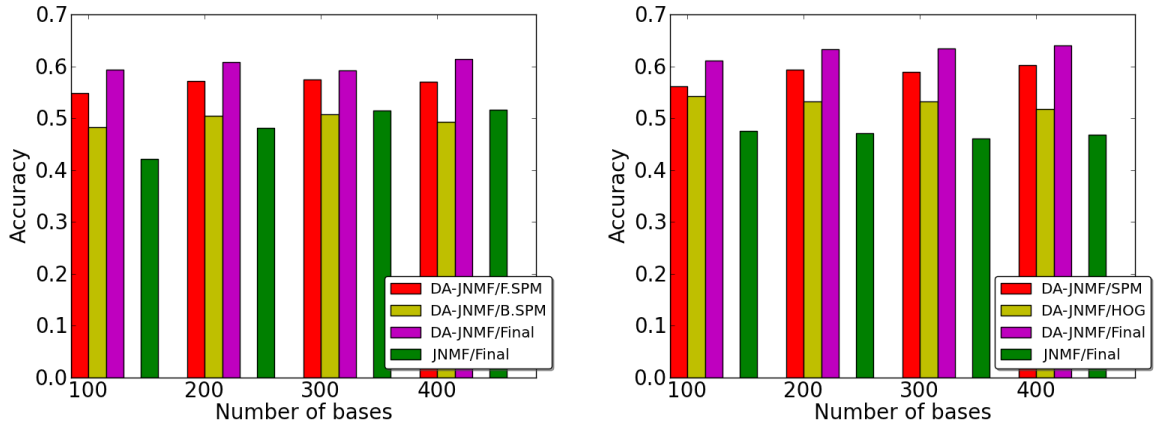
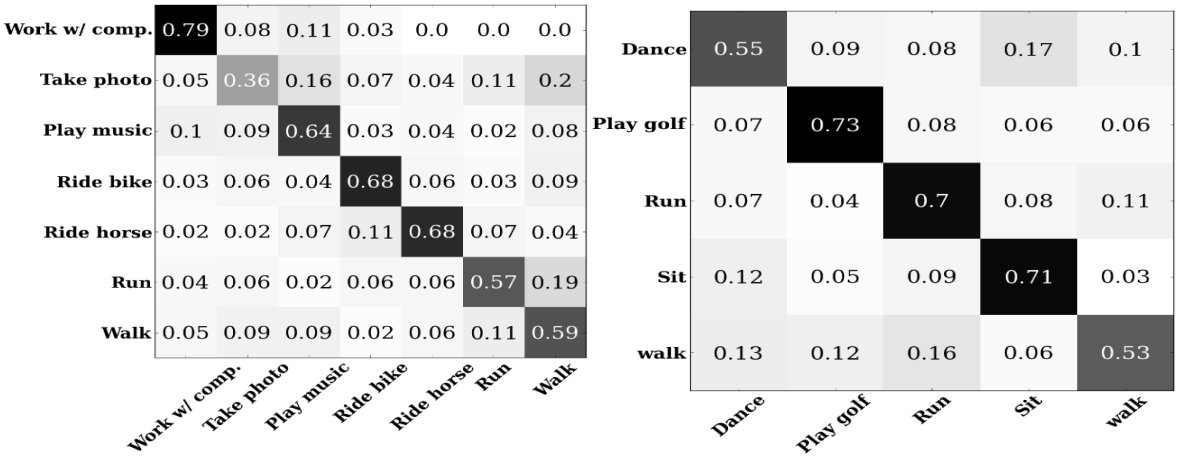


Figure 3.3: Classification accuracy for different number of bases K using the *HOG* and *SPM* features and (*JNMF*) compared to our proposed approach with *SPM* features (*DA-JNMF* (*DA-JNMF*/*SPM*), *HOG* features (*DA-JNMF*/*HOG*) and their fused results (*DA-JNMF*/Final) on the (a) *Willow* dataset, and the (b) *Web-actions* dataset

Table 3.1: Results on the *Willow* and the *Web-actions* datasets

Methods on Willlow	Overall acc. (%)	Mean per-class (%)
BOF+LSVM (Deltaire, Laptev and Sivic, 2010)	-	62.14
Our approach	61.04	61.57
Methods on Web-actions ds.		
Baseline (Ikizler, Cinbis and Sclaroff, 2009)	56.45	52.46
Latent Poses(Yang, Wang and Mori, 2010)	61.07	62.09
Our approach	**64.05**	**64.44**

scene appearance using the spatial pyramid. For the *Willow* dataset, our results still compare with state-of-the-art. An advantage of the proposed algorithm compared to state-of-the-art methods is in the classification model, which can be very efficient in real time applications as it requires only simple matrix multiplications, and summations. Finally, the confusion matrices of both datasets are depicted in Figure 3.4. Note that the major confusion within the *Web-actions* dataset occurs in the case of "dance" action with "sit", as both stands for different body pose articulations. Similarly, a noticeable confusion occurs between actions of "walk" and "run" as both have close pose and appearance views. For the *Willow* dataset, the action of "taking photo" is highly confused with other actions due to the limited visual clue of the presence of a camera for this action.



(a) (b)

Figure 3.4: Confusion matrices of our classification framework for the (a) *Willow* dataset, and the (b) *Web-actions* dataset.

3.6 Summary

In this chapter, we presented a novel classification algorithm based on recent advances in multiview *NMF*. The evaluations of this algorithm took over challenging action recognition datasets and demonstrated not only its significance for multiclass classification, but also its capability in benefiting from heterogeneous action features in a late fusion process. We showed also that the resulting classification model rely only on matrix multiplications in estimating classes posteriors, therefore, it represents a good candidate for real time applications where interest in classification confidence goes beyond one class to all other classes.

Despite the encouraging performance, our obtained recognition rates are not ideal for real world application scenarios due to the limited training data and the extreme inconsistency of the appearances of human actions. The typical inconsistency that exists in unconstrained action environments affects several defining patterns of the human actions such as actors appearances, action styles, camera views, scene appearance; making the classification using only appearance features a challenging task. Therefore, we investigate additional action representations that can provide better invariance towards variations in view, scale, and scene information while preserving the distinctive features among semantically different actions. In the following chapter, we elaborate further on a candidate representation that holds such properties. We also present a novel human action framework that better satisfies the key defining factors for human action recognition systems: robustness and efficiency.

CHAPTER 4

Discriminative Pose-based Framework for Human Action Recognition

4.1 Preface

Designing invariant action features against view, style, and scene variations is of great interest for reliable action recognition. Different actors and scenes often confer their own characteristics on the action representation, limiting the reliability of the extracted features. In Chapter 3, we presented a classification framework that only relied on appearance features. Despite the encouraging results in this domain, we discussed that several factors still greatly impede obtaining robust recognition performance using such representation. Among these factors were action variation across different viewing points, action styles, scales, and diverse appearances of actions' actors. Therefore, we resort to a high level action representation of human actions that subsides these inherited drawbacks of the appearance representation. This chapter presents a novel framework for human action recognition that is based on the pose representation of human actions. The proposed approach is based on a discriminative formulation of body-joints features that is: (i) invariant against different action styles, (ii) invariant against camera view variation, (iii) time efficient for both training and testing with low response latency, and (iv) achieves state-of-the-art results on four challenging human action

datasets of temporally segmented and unsegmented action videos. The benchmarks used for the evaluation in this chapter are *MSR-Action 3D* (Wanqing, Zhengyou and Zicheng, 2010b), *MSR-DailyActivity* (Wang et al., 2012a), *3D Action Pairs* (Oreifej and Liu, 2013), and *TUM Kitchen* (Tenorth, Bandouch and Beetz, 2009).

4.2 Introduction

Human action recognition has recently attracted an increasing interest in computer vision owing to its applications in many fields including surveillance, human computer interaction, and multimedia indexing. This interest resulted in a rapid development for the human action recognition in terms of its scale, algorithms efficiency, and action's input representation. Early approaches for action recognition assumed accurate measurements of the human poses (i.e. the spatial configuration of body joints) as an abstract high level representation of human actions. For instance, the approaches in (Campbell and Bobick, 1995; Bissacco et al., 2001; Ali, Basharat and Shah, 2007) use different phase space features of moving actor skeletons for action and gait recognition. However, in these days, obtaining an accurate measurements of the body pose often demanded special setups that are often time consuming and expensive.

Consequently, effort deviated toward alternative low and mid-level representations of pose, motion, visual appearance, or particular combinations of them for better action models. For instance, Wu, Oreifej and Shah (2011); Efros et al. (2003) rely majorly on motion cues to identify action sequences under static or moving camera setups. Our work presented earlier in Chapter 3, describe an action recognition approach that utilizes scene and body pose appearances to perform action recognition. Thurau and Hlavac (2008); Ikizler, Cinbis and Sclaroff (2009) harness the human pose appearance as the basic building block in discriminating actions. The introduction of interest points in video sequences (Laptev, 2005; Willems, Tuytelaars and Gool, 2008) led towards a successful adaption of the bag-of-words model for human action recognition (Laptev et al., 2008; Wang et al., 2011a; Xia and Aggarwal, 2013). Despite the encouraging results of low- and mid-level features for action recognition on several datasets, these approaches are greatly impeded by variations of view point, subject, scale, and appearance. Moreover, they lack a semantic meaning making the interpretation of the results sometimes difficult. In contrast, high-level representations (e.g., body pose) abstract most of these factors and provide a semantic interpretation of the results.

The recent advances in both human pose estimation algorithms and depth sensors have

rekindled interest in high-level human representations for action and behavior analysis (Ye et al., 2013). Despite their noisy estimations in monocular (Yang and Ramanan, 2013), depth sensors (Shotton et al., 2013), and multi-view (Yao, Gall and Gool, 2012) setups, several recent studies (Tran, Kakadiaris and Shah, 2011; Jhuang et al., 2013; Wang, Wang and Yuille, 2013; Yao, Gall and Gool, 2012) strongly point to the utility of pose estimation in obtaining superior or competitive performance as opposed to low- and mid-level features. Tran, Kakadiaris and Shah (2011) utilize polar coordinates of joints in a sparse reconstruction framework to classify human actions in realistic video datasets. Their evaluation clarifies the implication of accurate pose estimation on action recognition and identifies the potentials of current state-of-the-art pose estimation in obtaining excellent action recognition. Similar observations are reported in (Yao, Gall and Gool, 2012; Jhuang et al., 2013) on larger and more complex datasets. In particular, Jhuang et al. (2013) show that in some scenarios high-level features extracted by a current pose estimation algorithm (Yang and Ramanan, 2013) already outperform a state-of-the-art low-level representation based on dense trajectories (Wang et al., 2011a). These observations motivated researchers to further examine the potentials of high-level features under conventional and newly proposed challenges in videos and depth sensor data. For instance, Wang et al. (2012b) propose learning sets of most distinctive joints through mining. While Zanfir, Leordeanu and Sminchisescu (2013) weight poses of actions based on a mutual information criteria, Wang, Wang and Yuille (2013) mine for most occurring temporal and spatial structures of body joints for classification. A noticeable limitation in the aforementioned approaches resides in their demand of laborious mining of meaningful poses (Zanfir, Leordeanu and Sminchisescu, 2013), joints (Wang et al., 2012b), or temporal and spatial joints structures (Wang, Wang and Yuille, 2013), therefore, complicating model training and presenting considerable overhead on model future updates. Consequently, the applicability of these approaches for real world applications that demand online model learning with low latency is still questionable.

Unlike previous appraoches, we propose a pose-based algorithm for action recognition that is faster and more efficient for training and testing. Yet, it achieves on popular datasets for action recognition from 3D pose or RGB-D videos like (Wanqing, Zhengyou and Zicheng, 2010b; Oreifej and Liu, 2013), state-of-the-art performance and outperforms other related pose-based approaches. The efficiency is achieved by simplicity in design. Each joint is modeled by a single feature vector that encodes only the essential information to characterize an action: the relative location of the joint, the velocity of the joint, and the correlation

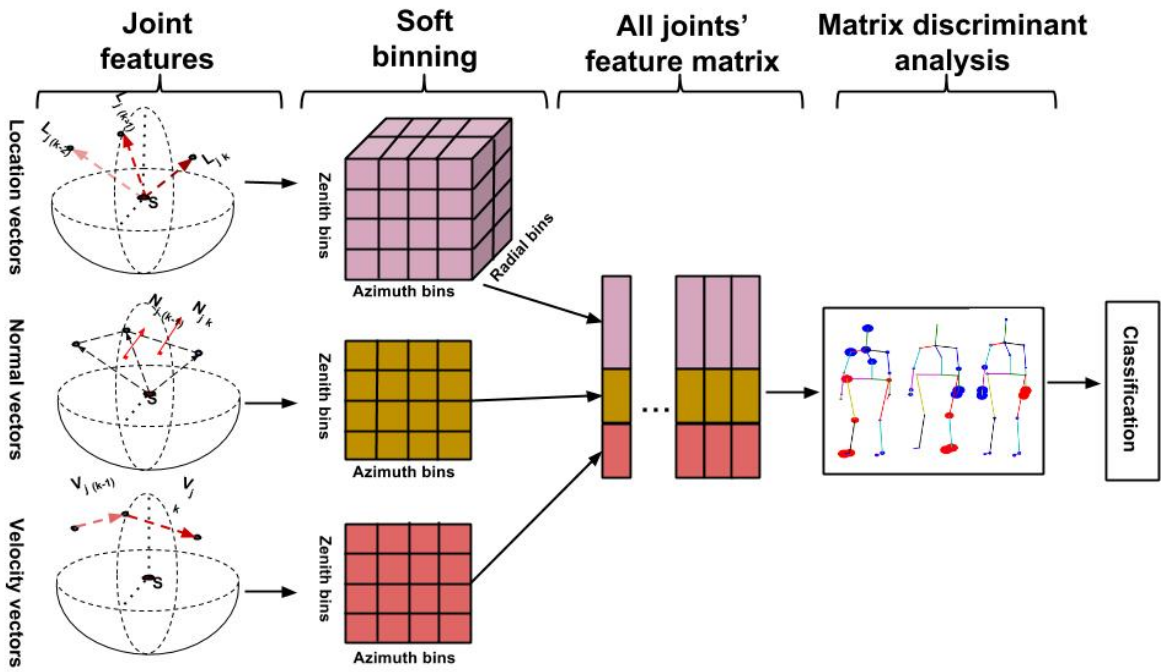
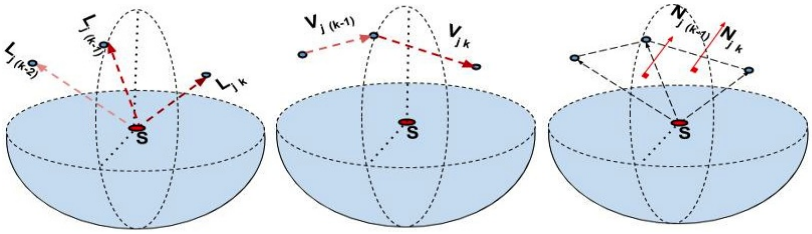


Figure 4.1: Overview of our pose-based framework for human action recognition.

between location and velocity. Inspired by Tran, Kakadiaris and Shah (2011), the information over a short video clip is encoded by histograms. Based on these features, a compact and discriminative representation is learned using partial least squares (PLS) Barker and Rayens 2003; Haj, Conzalez and Davis 2012; Harada et al. 2011; Schwartz et al. 2009; Sharma and Jacobs 2011. The representation can then be used with any classifier like SVM or Kernel-PLS (KPLS) Rosipal et al. 2001.

4.3 High-level Pose Representation

High-level pose representation presents the human pose as a collection of interconnected body parts and joints in a deformable configuration. To represent human actions by a high-level pose-based representation, a sequence of extracted 2D or 3D pose per frame is given. In order to be flexible and learn the importance of a single joint, our representation consists of a feature for each joint as depicted in Figure 4.1. The Joints features, which are discussed in Section 4.3.1 in more detail, model the distributions of the locations, velocities, and geometric orientation of the movements within a video clip or fixed number of frames as histograms. The histograms for each joint are then concatenated to build the feature matrix and matrix



(a) Location vectors (b) Velocity vectors (c) Normal vectors

Figure 4.2: Illustration of the locations feature f_l, velocities feature f_v, and the normals feature f_n for a single joint j. For each frame k or frame pair $(k, k+1)$, the vectors l_{jk}, v_{jk}, and n_{jk} are converted into spherical coordinates and added to a histogram as shown in Figure 4.1.

discriminant analysis is performed to obtain a set of discriminant eigenvectors, which are used as high-level representation of the video clip. The representation can then be used with any classifier for classification.

4.3.1 Joint Features

To increase the robustness of the features to variations caused by different body shapes or even foreshortening in case of 2D pose, we convert relative joint positions and other vectors into a spherical coordinate system. 2D vectors from 2D poses are represented by the length r and the orientation angle $\theta \in [0, 360]$. For a 3D skeleton representation, we use the horizontal orientation or azimuth $\alpha \in [0, 360]$ and the vertical orientation or zenith $\phi \in [0, 180]$. A vector $v = (x, y, z) \in \mathcal{R}^3$ is then converted into spherical coordinates (r, α, ϕ) by:

$$r = \sqrt{x^2 + y^2 + z^2} \tag{4.1}$$

$$\phi = \frac{180}{\pi} \times (\arccos\left(\frac{z}{r}\right)) \tag{4.2}$$

$$\alpha = \frac{180}{\pi} \times (\text{atan2}(y, x) + \pi) \tag{4.3}$$

Using spherical coordinates, we propose three features that represent distributions over a fixed set of K frames as 3D or 2D histograms. For each feature, we indicate if it applies to 2D and 3D poses or both:

Joint Location Feature f_l (2D and 3D):

The f_l features resemble their 2D counterparts presented in (Tran, Kakadiaris and Shah, 2011). But with 3D skeletons, our representation includes the azimuth α and zenith ϕ angles along with the joint displacement r from a reference point. The reference point is selected as the location of the *spine s*, which naturally corresponds to the center of the body. For a given location x_{jk} of a joint j at frame k, we quantize the polar coordinates (r, α, ϕ) of the joint location vector $l_{jk} = x_{jk} - s$ into a 3D histogram $(R \times O_{lv} \times O_{lh})$, where R, O_{lv}, O_{lh} are the number of bins for radius, vertical, and horizontal angle. The location vectors of all frames but of a single joint are accumulated in a single 3D histogram. The joint location vectors for three frames and one joint are illustrated in Figure 4.2 (a). Thus, the locations feature f_l consists of J 3D histograms, where J is the number of joints. In case of 2D pose, the histograms are 2D corresponding to the 2D coordinates (r, θ) for each 2D vector.

Joint Velocity Feature f_v (2D and 3D) :

The joint location features do not encode any temporal information. The joint motion is, however, of great significance for understanding the semantics of actions, especially for actions consisting of identical body and joints configuration but in different temporal order such as carry/put; stand-up/sit-down; and catch/throw. Given the locations of a joint j at successive frames l_{jk} and $l_{j(k+1)}$, we convert the velocity vector $v_{jk} = l_{j(k+1)} - l_{jk}$ into spherical coordinates without radius (α, ϕ). The radius is not taken into account to be invariant to different execution speeds of an action among subjects. The velocity vectors for all $K-1$ frame pairs are then added to the 2D histogram $O_{vv} \times O_{vh}$, where O_{vv} and O_{vh} are the numbers of bins for vertical and horizontal angle. The velocity vectors for two frame pairs are illustrated in Figure 4.2 (b). The velocities feature f_l therefore consists of J 2D histograms. The features f_l are in many cases complimentary to the f_v features. While f_v captures the velocity distributions of all joints, f_l captures the location distributions of all joints. This is important for actions where there is not much movement for some joints, but their relative position matters for the interpretation (e.g. call cellphone,sit still and write on a paper).

Joint Movement Normals Feature f_n (3D only):

Joint locations feature f_l and joint velocities feature f_v treat joint locations and joint velocities independently. The joint movement normals feature models the correlation of location and

velocity. To this end, the cross product between the location vector l_{jk} and the velocity vector v_{jk} is computed or more efficiently $n_{jk} = l_{jk} \times l_{j(k+1)}$. Up to a scaling factor, n_{jk} corresponds to the normal of the plane spanned by the three points s and the joint positions at the two frames k and $k+1$. Since the length of the normal vector is one, we convert n_{jk} into spherical coordinates (α, ϕ) without r. The normals of the $K-1$ frames are quantized as the velocities feature into a 2D histogram and we obtain J 2D histograms for f_n. The movement normals for two frame pairs are illustrated in Figure 4.2 (c). All three features model only the most essential information to characterize an action: the relative locations of the joints, the velocities of the joints, and the correlations between locations and velocities. However, combined with a discriminative approach to learn a basis for the features, which is described in detail in Section 4.3.2, we achieve state-of-the-art performance and outperform features that are much more expensive to compute.

Impact of Feature Quantization

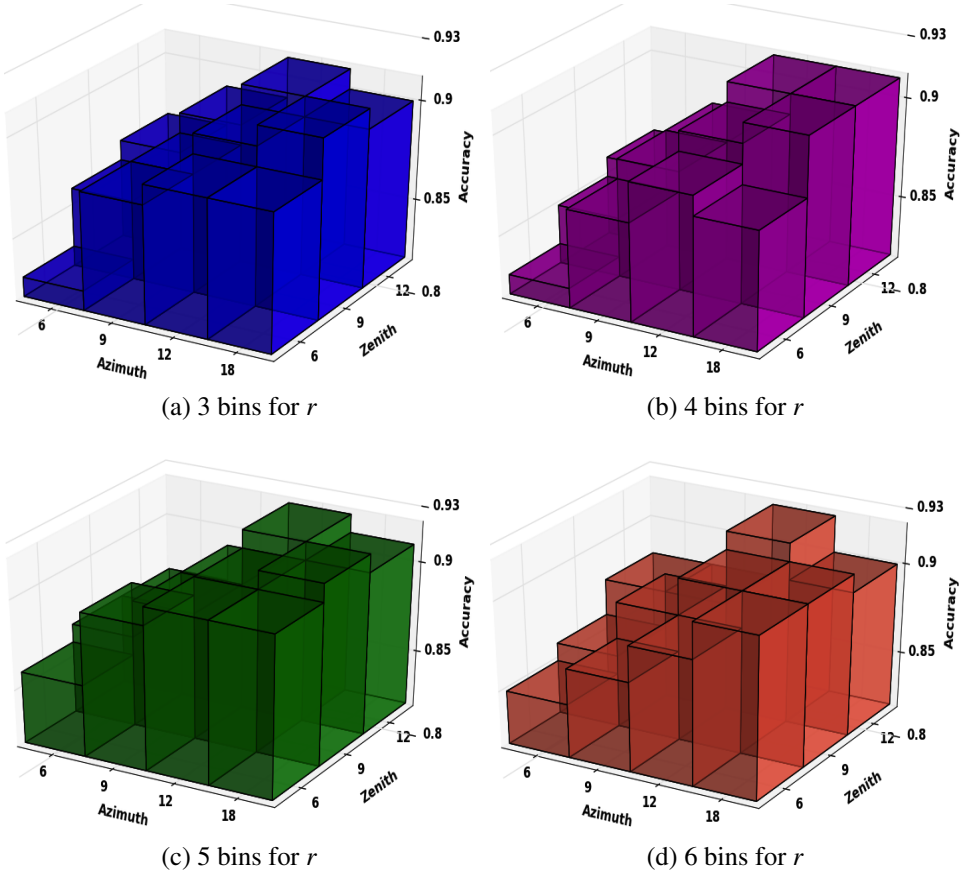
To evaluate the impact of feature quantization, we measured the average classification accuracy over three different splits for *MSR-Action3D* for various quantization of length r, azimuth α, and zenith ϕ of the joint locations, joint velocities, and joint movement normals. The results are shown in Figure 4.3. While several configurations give a good performance, we chose 5, 18, and 9 as the number of bins for length, azimuth, and zenith, respectively. This configuration is used for all the experiments presented later in this chapter.

Normalization and Soft-binning

To reduce any binning artifacts and to be more robust against style variations, we perform soft-binning. This is achieved by adding a quantized vector to all neighboring bins. The weights for the bins are given by a Gaussian kernel with $\sigma = 1$. To handle sequences of different lengths, the histograms are normalized by the L2-norm.

Temporal Pyramid

In addition, a temporal pyramid can be used. Instead of having a single histogram per video clip, it can be subdivided into smaller temporal segments. Since the videos in the datasets are short, we use a pyramid with only two layers. The second layer divides the video in three equally sized parts. The three histograms of the second layer and the histogram of the first



(a) 3 bins for r (b) 4 bins for r

(c) 5 bins for r (d) 6 bins for r

Figure 4.3: Recognition accuracy for different feature quantizations for length r, azimuth α, and zenith ϕ. The plots show the accuracy when the number of bins changes. There are several configurations that give a good performance. Among them we use 5, 18, and 9 as the number of bins for length, azimuth, and zenith, respectively.

layer are then concatenated.

4.3.2 Learning Discriminative Action Features

Human actions perception is closely tied with our semantical interpretation of body joints articulations. These articulations may vary significantly across different actors and styles for the same action. For example, consider the snapshot of the "hammering" action in Figure 4.4. While the poses vary significantly across those samples, we still interpret their action as "hammering" putting extra weights on the movements of one hand and neglecting all irrelevant articulation of the rest of the body. This intuition can be formally expressed by a weighting scheme on top of the features for each joint. Given a set of J joint features

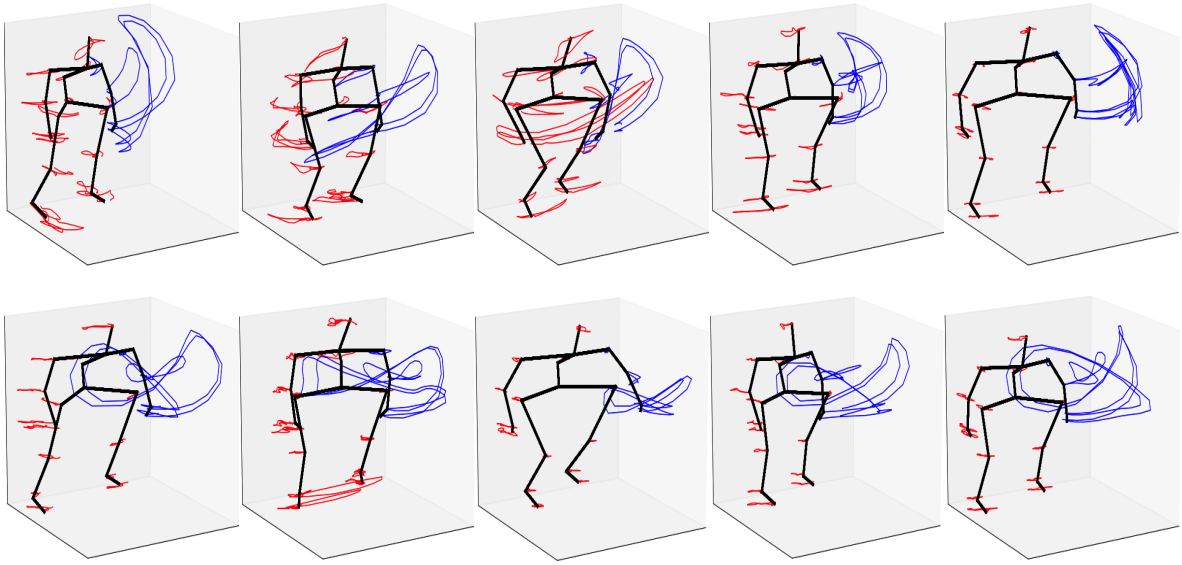


Figure 4.4: Examples of joint's trajectories for the hammering (first row) and (draw X) actions from *MSR-Action3D* dataset (Wanqing, Zhengyou and Zicheng, 2010b). Relevant action's trajectories of left hand, left wrist, and left elbow are marked in *blue*, while other inconclusive trajectories are marked in *red*. Notice that both vary significantly across different actions and actors posing more challenges in representing human action.

$\{f_j \in \mathbb{R}^D\}_{j=1}^J$ that capture their underlying semantics. We define the pose feature $\mathbf{f_p} \in \mathbb{R}^D$ as a weighted sum of its joints by the following equation:

$$\mathbf{f_p} = \sum_{j=1}^{J} w_j f_j \tag{4.4}$$

which can be expressed in matrix form as:

$$\mathbf{f_p} = \mathbf{Fw} \tag{4.5}$$

where columns of $\mathbf{F} \in \mathbb{R}^{(D \times J)}$ corresponds to the joints' features as illustrated in Figure 4.1 and $\mathbf{w} \in \mathcal{R}^J$ defines their corresponding weights. As the joints' features $\mathbf{F}$ has a matrix based representation, learning suitable weights $\mathbf{w}$ can be approached through a matrix based discriminant analysis scheme (Li and Yuan, 2005; Barker and Rayens, 2003; Bauckhage and Kaster, 2006; Harada et al., 2011). In this work, we investigate two approaches: 2D-LDA and PLS.

2D-LDA

2D-LDA (Li and Yuan, 2005) is a discriminant linear analysis method that operates on 2D matrices. Let M denote the number of training videos and K the number of classes. We then have for each class k, M_k videos and extract from all videos the feature matrices $\{\mathbf{F_i}\}_{i=1}^{M}$. We further compute the mean of the training videos, denoted by $\overline{\mathbf{F}}$, and the means for each class, denoted by $\{\overline{\mathbf{F_k}}\}_{k=1}^{K}$.

The idea of 2D-LDA is to search for a projection vector w such that the combined pose features f_p for different classes have small within-classes and large inter-class variation. This is achieved by the weighting vector w that maximizes the fisher criterion

$$J(x) = \frac{\operatorname{tr} \mathbf{S_b}}{\operatorname{tr} \mathbf{S_w}}, \tag{4.6}$$

where $\mathbf{S_b}$ and $\mathbf{S_w}$ can be written as follows:

$$\mathbf{S_b} = \frac{1}{M} \sum_{k=1}^{K} M_k \left[(\overline{\mathbf{F_k}} - \overline{\mathbf{F}})w \right] \left[(\overline{\mathbf{F_k}} - \overline{\mathbf{F}})w \right]^T \tag{4.7}$$

$$\mathbf{S_w} = \frac{1}{M} \sum_{k=1}^{K} \sum_{i_k \in M_k} \left[(\mathbf{F_{i_k}} - \overline{\mathbf{F_k}})w \right] \left[(\mathbf{F_{i_k}} - \overline{\mathbf{F_k}})w \right]^T . \tag{4.8}$$

This can be rewritten as $\operatorname{tr} \mathbf{S_b} = w^T \mathbf{\Sigma_b} w$ and $\operatorname{tr} \mathbf{S_w} = w^T \mathbf{\Sigma_w} w$ where

$$\mathbf{\Sigma_b} = \frac{1}{M} \sum_{k=1}^{K} M_k \left[(\overline{\mathbf{F_k}} - \overline{\mathbf{F}}) \right]^T \left[(\overline{\mathbf{F_k}} - \overline{\mathbf{F}}) \right] \tag{4.9}$$

$$\mathbf{\Sigma_w} = \frac{1}{M} \sum_{k=1}^{K} \sum_{i_k \in M_k} \left[(\mathbf{F_{i_k}} - \overline{\mathbf{F_k}}) \right]^T \left[(\mathbf{F_{i_k}} - \overline{\mathbf{F_k}}) \right]. \tag{4.10}$$

Estimating for the optimal weighting vector w^* therefore simplifies to solving a generalized eigenvalue problem:

$$\mathbf{\Sigma_b} w^* = \lambda \mathbf{\Sigma_w} w^* \tag{4.11}$$

In practice, however, $\mathbf{\Sigma_w}$ can be often singular, specially in cases where the number of training samples is less than the feature dimension for the joints.

PLS

PLS has been recently adopted in computer vision for different applications including pose estimation and regression (Haj, Conzalez and Davis, 2012), image classification(Harada et al., 2011), pedestrian detection (Schwartz et al., 2009), and multi-view learning (Sharma and Jacobs, 2011). The PLS algorithm aims at extracting common information between two sets $\mathcal{X} = [x_1, \cdots, x_m]_{n\times m}$ and $\mathcal{Y} = [y_1, \cdots, y_m]_{n\times m}$ by learning a set of orthogonal linear projections a, b that maximize the following criteria:

$$\underset{a^T A=0}{\operatorname{argmax}} \left\{ \frac{[cov(a^T x, b^T y)]^2}{(a^t a)(b^T b)} \right\} \tag{4.12}$$

According to Barker and Rayens (2003); Harada et al. (2011), an estimation of the weight vector a reduces to the eigenvalue problem of the between-class covariance matrix $\mathbf{S_b}a = \lambda a$ where $\operatorname{tr}\mathbf{S_b} = w^T\mathbf{\Sigma_b}w$. By maximizing $\mathbf{S_b}$ under the condition of $w^T w = 1$ we obtain the weighting vector w^* as the eigenvector that corresponds to the highest eigenvalue of the following eigenvalue problem:

$$\mathbf{\Sigma_b}w^* = \lambda w^* \tag{4.13}$$

For both 2D-LDA and PLS, we chose an adequate number of eigenvectors corresponding to the largest P eigenvalues. Hence, the final feature f_p of an video sample i with feature matrix $\mathbf{F_i}$ is given by $f_p = [\mathbf{F_i}w_1, \mathbf{F_i}w_2, \cdots, \mathbf{F_i}w_P]$ where $f_p \in \mathbb{R}^{(D\times P)}$.

Figure 4.5 depicts the first seven eigenvectors learned using PLS on the *MSR-Action3D* and *DailyActivity* datasets. Blue and red signify positive and negative weights respectively, and the size of the circles refer to the absolute value of the joint weight. Notice that most of the eigenvectors focus on joints that are relevant to discriminate between actions. For instance, in *MSR-Action3D* dataset, only a few body part combinations are relevant where some joints like the hips are irrelevant for the human actions.

4.3.3 Classification

The obtained action features $\mathbf{f_p}$ can be classified using any off-the-shelf classifier like SVM. In our experiments, we use a non-linear classifier based on PLS, namely Kernel-PLS (KPLS) (Schwartz et al., 2009; Rosipal et al., 2001). As training data, we have for each video clip the label and the feature vector $\mathbf{f_p}$ which are transformed so that all its entries are positive. While

the features define the set $\mathcal{X}$, the class labels are encoded by the set $\mathcal{Y}$. As kernel, we use the intersection kernel defined as $K_{i,j} = \sum_l \min\left(\mathbf{f}_{\mathbf{p_i}}(l), \mathbf{f}_{\mathbf{p_j}}(l)\right)$.

4.4 Datasets and Experiments

We choose four challenging datasets to evaluate our approach for human action recognition. The datasets are *MSR-Action3D* (Wanqing, Zhengyou and Zicheng, 2010b), *3D Action Pairs* (Oreifej and Liu, 2013), *MSR-DailyActivity3D*[1], and *TUM Kitchen* dataset (Tenorth, Bandouch and Beetz, 2009). For all the experiments in the following sections, we used the same parameters (number of bins) to construct our pose features as discussed earlier in Section 4.3.1. We used 18 bins for the horizontal orientation (azimuth) and nine bins for vertical orientation (zenith) for 3D skeletons. While we used 18 bins for encoding the orientation angle (θ) in 2D. Our experiments are performed on the provided pose data that has been captured using the *Kinect* skeletal tracker for the first three datasets and by a model-based approach for the *TUM* dataset. On all experiments, we learn the classifier parameters using 5-fold cross validation.

4.4.1 MSR-Action3D

The MSR-Action3D dataset is an action dataset captured with a RGB-D camera and designated for gaming-like interactions. It consists of 567 temporally segmented action sequences and contains 20 actions, each performed two to three times by ten different subjects. The actions are: “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”. We exclude ten sequences as in (Wang et al., 2012b) and operate on the X,Y screen coordinates along with their corresponding depth.

For evaluation, we follow the work in (Wang et al., 2012b; Oreifej and Liu, 2013) and consider two evaluation tasks: (i) The cross-subject setup where we train our model using the actions of subjects 1, 3, 5, 7, 9 and report the results on the rest. (ii) The second task reports the system performance on the average accuracy on all **252 (5-5)** cross-subject splits.

[1] http://research.microsoft.com/en-us/um/people/zliu/ActionRecoRsrc/

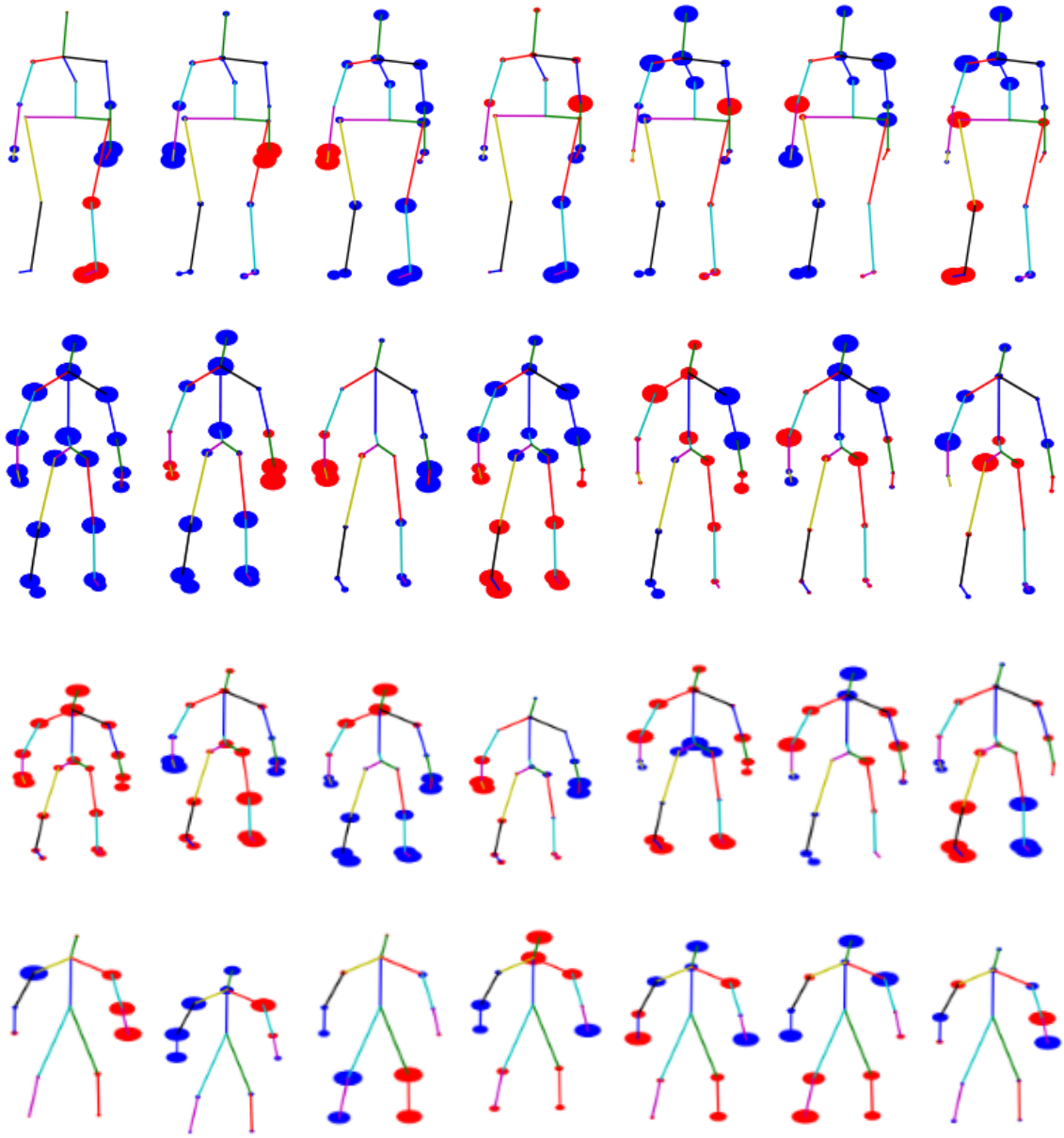
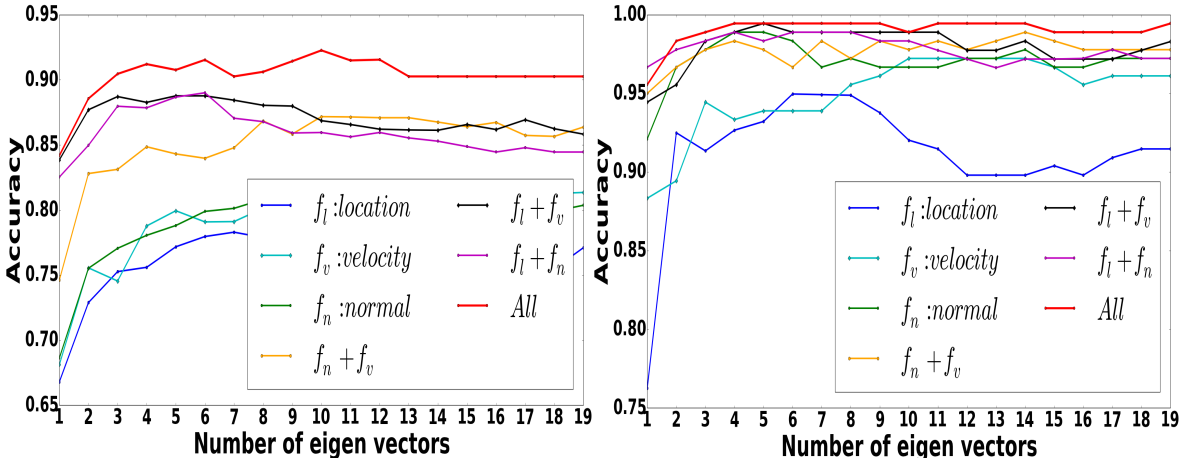


Figure 4.5: The first seven discriminative projections of joint's features extracted using PLS from *MSR-Action3D* (first row), *DailyActivity* datasets (second row), 3D actions pairs dataset (third row) and TUM dataset (fourth row). Notice that only few part combinations in *MSR-Action3D* dataset are relevant where other joints like the hips are irrelevant for human actions. While in *TUM*, mostly the upper parts joints features are important as the actions of this dataset correspond to the daily human actions performed in a kitchen. Red and blue colors signify negative and positive weights respectively, while the size of the joint signifies its weight.



(a) MSR-Action3D (b) 3D Action Pairs

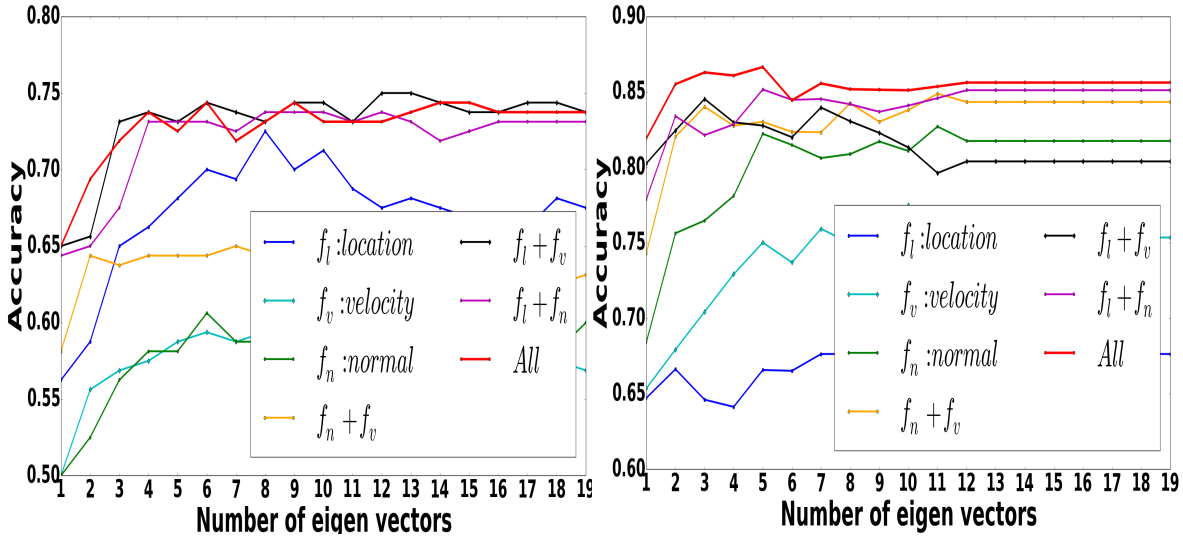
Figure 4.6: Recognition accuracies for different numbers of eigenvectors and various feature combinations for (a) *MSR-Action3D* and (b) *3D Action Pairs* datasets.

Using the first task, Figure 4.6 (a) shows the individual contribution of each joint feature with respect to the number of projection vectors obtained by PLS. The combinations of the three features f_l, f_v, and f_n, which capture joint location, velocity, and their correlation, clearly boost the performance in comparison to each single feature or feature pairs. Using all three features and 10 pls-projections, an accuracy of **92.3**% is achieved.

We further evaluated the impact of soft-binning in Figure 4.8 (a). Without soft-binning the descriptor is more sensitive to style variations and binning artifacts. Soft-binning therefore improves the results by a margin.

To evaluate the impact of feature quantization, we measured the average classification accuracy over three different splits for *MSR-Action3D* for various quantization of length r, azimuth α, and zenith ϕ of the joint locations, joint velocities, and joint movement normals. The results are shown in Figure 4.3. While several configurations give a good performance, we chose 5, 18, and 9 as the number of bins for length, azimuth, and zenith, respectively. This configuration is used for all datasets.

Figure 4.8 (b) compares 2D-LDA with PLS. Due to singularities the performance drops for 2D-LDA when adding more eigenvectors. In contrast, PLS does not suffer from singularities. However, both approaches perform better than just concatenating the features and using a SVM. The difference in performance between KPLS and SVM trained both using an intersection kernel is shown in Figure 4.8 (c). While for few eigenvectors the performance is the same, KPLS improves with more eigenvectors in contrast to the SVM.



(a) (b)

Figure 4.7: Recognition accuracies for different numbers of eigenvectors and various feature combinations for MSRDailyActivity *MSRDailyActivity* and *TUM Kitchen* dataset

Table 4.1 compares our approach with the state-of-the-art on this dataset. Our approach achieves an accuracy of **92.3**%. It performs comparable to the state-of-the-art (Wang and Wu, 2013) and performs better than other skeleton-based approaches. The temporal pyramid does not improve the results since the dataset contains short, well-defined actions where temporal invariance is beneficial. Figure 4.9 shows the confusion matrices of our predictions for *MSR-Action3D* with and without temporal pyramiding. There are only few actions where the accuracy is not very high, namely, "hand catch", "high throw", "draw x", and "pickup and throw". Without the temporal pyramid, the action of "draw x" is confused with "hammer" since the movements can be very similar when the features are invariant to the magnitude of the velocity. With the temporal pyramid, "draw x" is distinguished from "hammer", but it is confused with "draw circle" and "draw tick" since all three activities share very similar movements for temporal sub-parts of the actions.

To verify the invariance of our features against different subjects. We evaluate our algorithm against all possible 5–5 subjects splits of the data. In total this ends up with **252 (5-5)** possible splits. For this task we achieved a mean accuracy of **88.38**% and standard deviation of **0.027** that is of margin better than current state-of-the-art results of **82.15**%±**4.18** in (Oreifej and Liu, 2013). This provides an empirical evidence of the method's robustness against cross-subject variations for human action recognition.

Table 4.1: Recognition accuracy for *MSR-Action3D* dataset. The methods use different data modalities where **S** denotes skeleton data, **D** depth, and **TP** denotes the use of a temporal pyramid.

Method	Modality	Accuracy(%)
(Wang and Wu, 2013)	**D+S**	92.76
(Wang et al., 2012b)	**D+S**	88.2
(Wang et al., 2012c)	**D**	86.5
(Oreifej and Liu, 2013)	**D**	88.36
(Zanfir, Leordeanu and Sminchisescu, 2013)	**S**	91.7
(Wang, Wang and Yuille, 2013)	**S**	90.22
(Xia and Aggarwal, 2013)	**D**	89.3
Ours	**S**	**91.5**
Ours(TP)	**S**	**90.1**

For the runtime evaluation, we estimated the training and testing time on the *MSR-Action3D* standard split to be 27 and 14 seconds respectively. More precisely, the classification time required for a video clip comprised of 55 frames is 161ms where the feature extraction step takes 148ms. The approach presented by **myZanfir** provides comparable results in terms of classification time, however, their training time is much more expensive since each frame is classified by a kNN classifier. We also compared with the recent approach presented by (Vemulapalli, Arrate and Chellappa, 2014), which uses DTW and requires many mappings between Lie group and tangential space. Using the provided source code on the same machine, we found that classifying a single video clip of 58 frames requires around 20 seconds. All the experiments were conducted on an Intel Core i7 CPU, 3.40GHz machine with an 8Gbyte RAM. This shows that our approach is both very efficient for training and testing.

4.4.2 3D Action Pairs Dataset

This dataset emphasizes on particular scenarios where motion and shape cues are highly correlated. It comprises of six pairs of actions, such that within each pair the motion and the shape cues are similar, but their temporal correlations vary. The action pairs are: “Pick up a box/Put down a chair”, “Lift a box/Place a box”, “Push a chair/Pull a chair”, “Wear a hat/Take off hat”, “Put on a backpack/Take off a backpack”, and “Stick a poster/Remove a poster”. We evaluate our framework using the same cross-subject evaluation protocol as in *MSR-Action3D*.

Figure 4.6 (a) shows the individual performance of each feature for different projections.

Table 4.2: Recognition accuracy for 3D Action Pairs. The methods use different data modalities where **S** denotes skeleton data, **D** denotes depth, and **TP** denotes the use of a temporal pyramid.

Method	Modality	Accuracy(%)
(Wang and Wu, 2013)	**D+S**	97.22
(Wang et al., 2012b)	**D+S**	82.22
(Oreifej and Liu, 2013)	**D**	96.67
Ours	**S**	**92.0**
Ours(TP)	**S**	**99.4**

For this datasets, the correlation features f_n outperform the location and velocity features since they capture temporal-spatial correlations of the action classes better. The best performance is, however, achieved when all features are used.

We compare our approach with the state-of-the-art in Table 4.2. Our algorithm achieves **93.09**%. When a temporal pyramid is used, it achieves **99.4**% and outperforms the other methods. The performance boost of the pyramid can be explained by the classes. These are activities that consist of smaller sub-actions in a specific order, which can be well captured by the temporal pyramid.

Figure 4.10 shows the confusion matrix for *3D Action Pairs*. In contrast to *MSR-Action3D*, the temporal pyramid enhances the classification accuracy for *3D Action Pairs*. Without using a temporal pyramid, actions with high temporal correlations are confused (i.e. *place box* and *lift box*). The temporal pyramid resolves this confusion by being able to distinguish the order of the motion that is affected by the box.

4.4.3 MSRDailyActivity

This dataset has been captured with an RGB-D camera to mimic daily human activities in a living room. There are 16 different actions, each performed by ten subjects twice, once standing and the other while sitting. The actions are: “drink”, “eat”, “read book”, “call cellphone”, “write on a paper”, “use laptop”, “use vacuum cleaner”, “cheer up”, “sit still”, “toss paper”, “play game”, “lie down on sofa”, “walk”, “play guitar”, “stand up”, “sit down”. The standard task for this dataset aims at cross subject evaluation as in *MSR-Action3D*, where we train on the odd numbered subjects and test on the rest.

Figure 4.7 (a) shows the individual accuracies of the different features. Unlike *MSR-Action3D* and *3D Action Pairs* datasets, the joints location feature (f_l) in this dataset outper-

forms both velocity and normal features. This is because many actions in this dataset are of static or merely static nature (e.g.,"call cellphone", "play game", "use laptop"). However, our combined features outperform the individual features and achieve an overall accuracy of **70.0**%. With a temporal pyramid, the accuracy is further improved to **73.1**% accuracy.

Compared to previous work (Wang et al., 2012b), our method outperforms their results of **68.0**% by **6.75**%. Figure 4.11 depicts the confusion matrices obtained using our approach with and without temporal pyramiding. Notice that temporal pyramiding alleviates the confusion between different actions of high temporal correlations. However, the recognition is still confused for actions with little movement like "write", "use laptop", or "call cell phone". For these actions, the involved objects need to be taken into account to improve the performance.

4.4.4 TUM Kitchen Dataset

The *TUM kitchen* dataset focuses on a home-monitoring scenario using a multi-view camera setup (4 cameras). The dataset provides 3D human pose data estimated by a markerless full-body tracker. Our evaluation criteria consider two tasks: (i) *segmented* test data and (ii) *unsegmented* test data as in (Yao, Gall and Gool, 2012). On both tasks, we used the episodes 0-2,0-8,0-4,0-6,0-10,0-11,1-6 for testing and the remaining 13 for training. However, in the first task we assume that the videos are already segmented while in the second task we perform continuous classification. The evaluation criteria for the unsegmented case follow the protocol described in (Yao, Gall and Gool, 2012), where the average class accuracies is measured on a frame-level. We use the skeleton with 13 joints for evaluation and do not count the errors at the transition frames between annotations with a margin of 4 frames on both sides as in (Yao, Gall and Gool, 2012). For the first task, Figure 4.7 (b) presents a detailed overview of the average recognition accuracies over all classes for each feature along with their combination. Our algorithm achieves for this task an average accuracy of **86.65**% over all classes.

On the second task, we evaluated the performance of our approach using a fixed sliding window of 30 frames that was determined empirically. This task is more challenging as the dataset stands for actions of arbitrary time stamps ranging from 10 to 150 frames. The evaluation considers the average accuracy on frame level over all classes. Our algorithm achieves an average accuracy of **82.5**% as compared to **80.03**% in (Yao, Gall and Gool, 2012). Figure 4.12 (a) depicts the prediction of our model for an unsegmented action sequence from the *TUM* dataset. While Figure 4.12 (b) shows the confusion matrix for all classes.

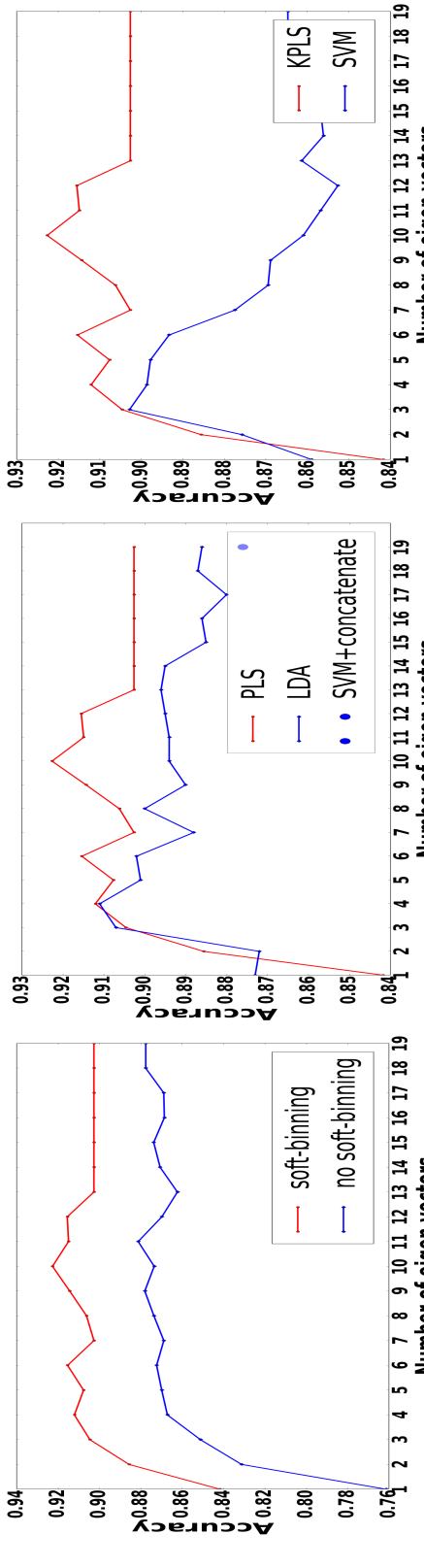
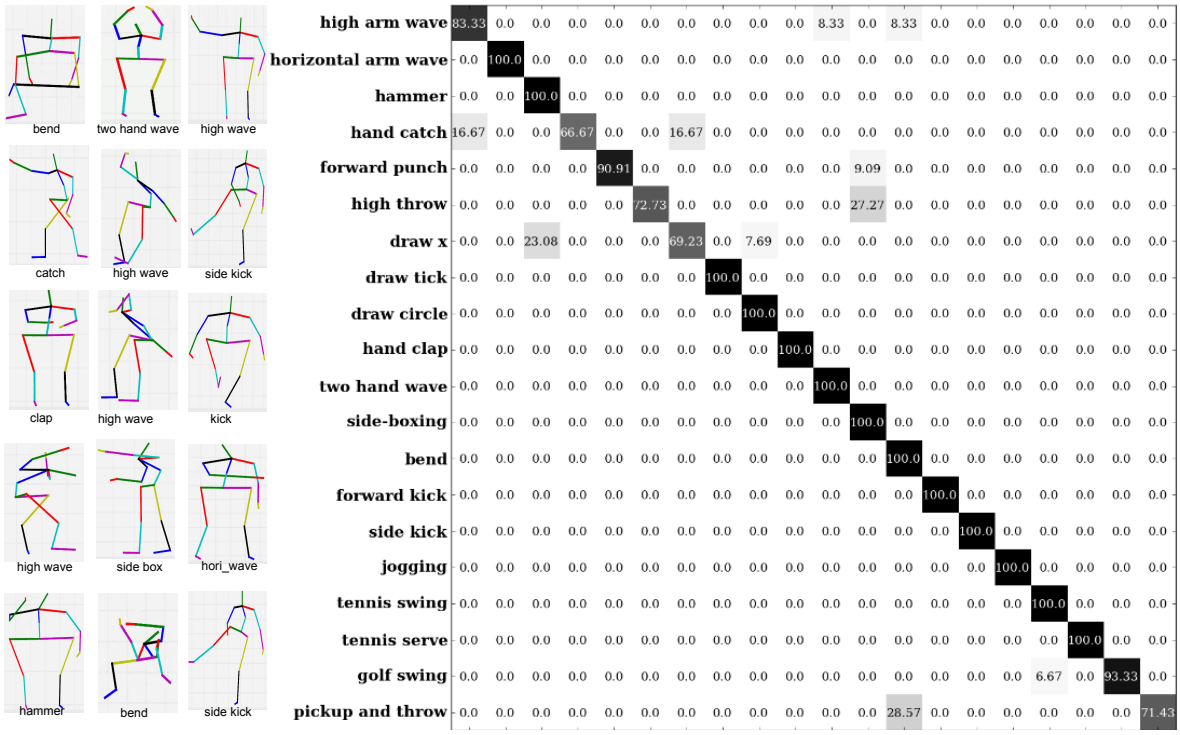
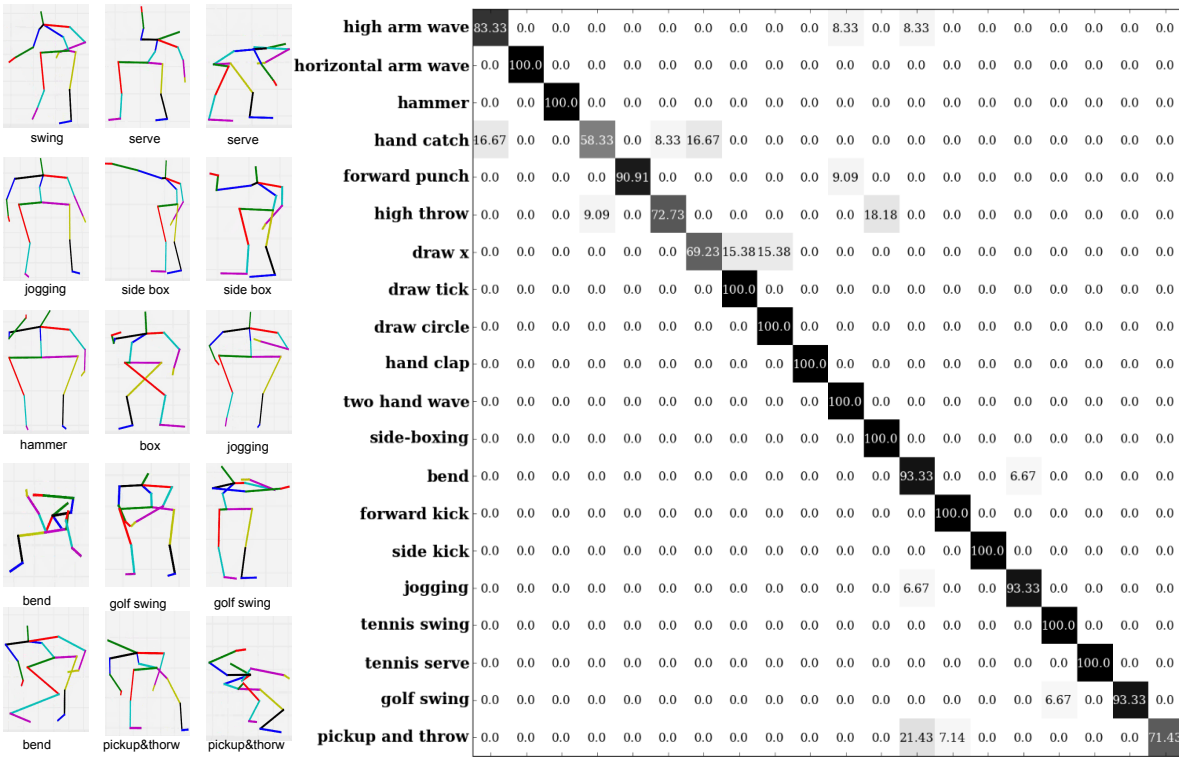


Figure 4.8: Performance evaluation on *MSR-Action3D* dataset: (a) impact of soft binning, (b) comparison of 2D-LDA and PLS, and (c) comparison of KPLS and SVM classifiers

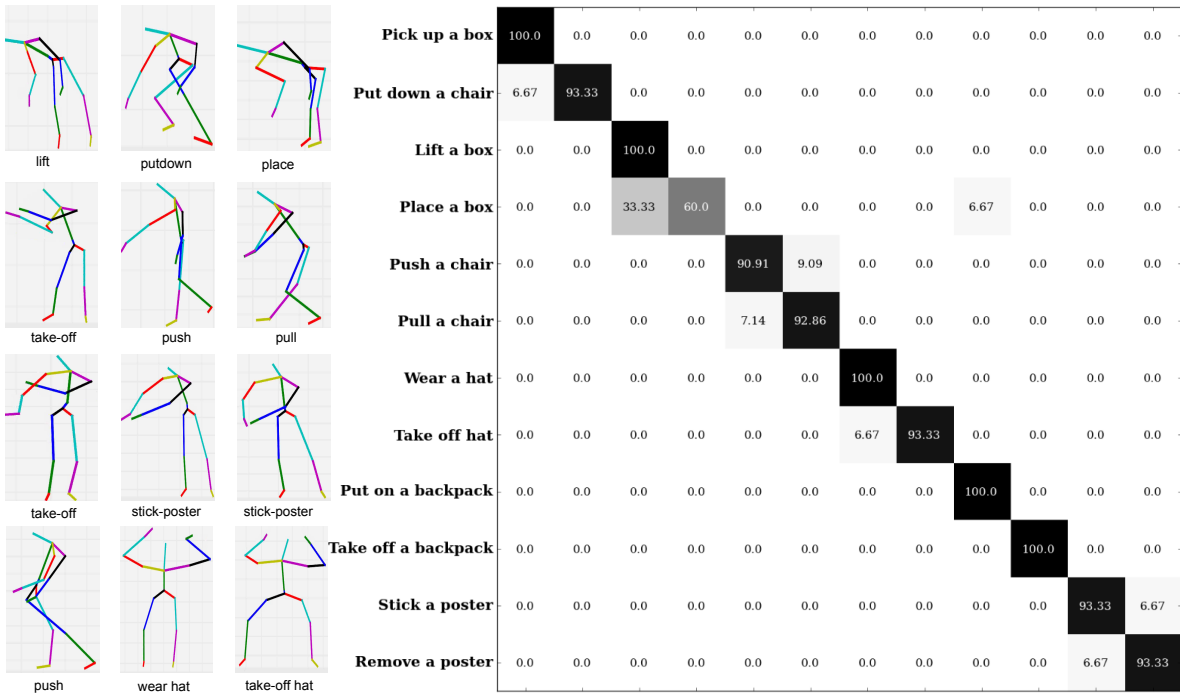


(a) Without temporal pyramid

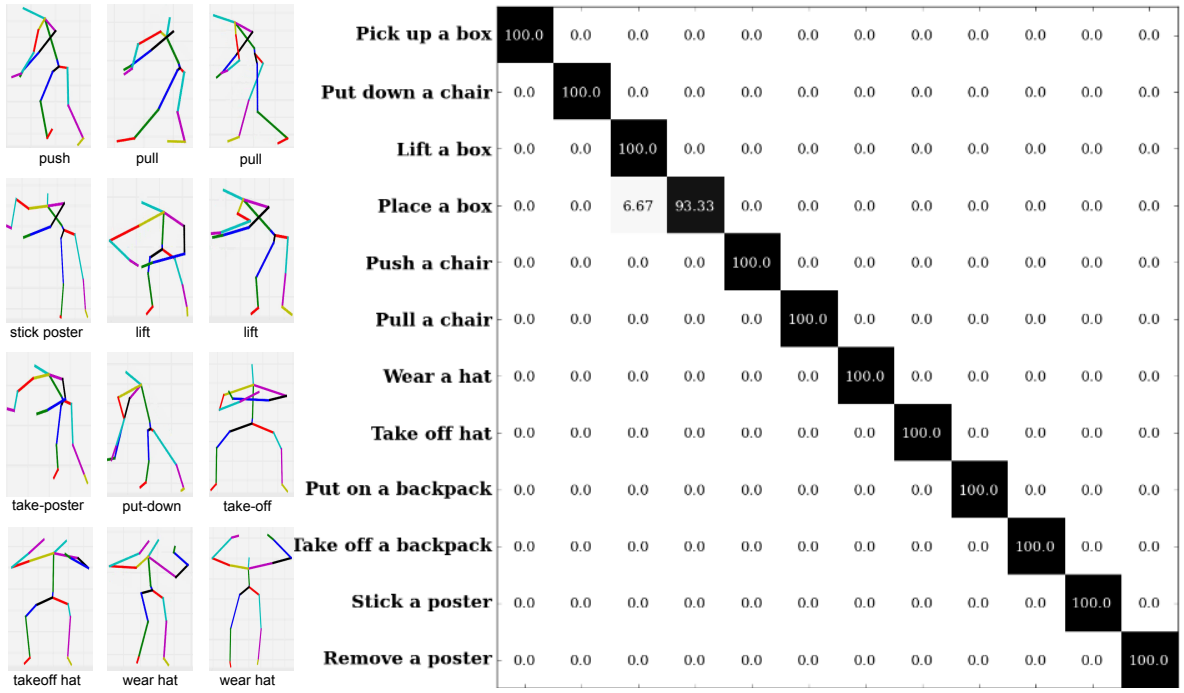


(b) With temporal pyramid

Figure 4.9: Confusion matrices for *MSR-Action3D* obtained (a) without a temporal pyramid, and (b) with a temporal pyramid.

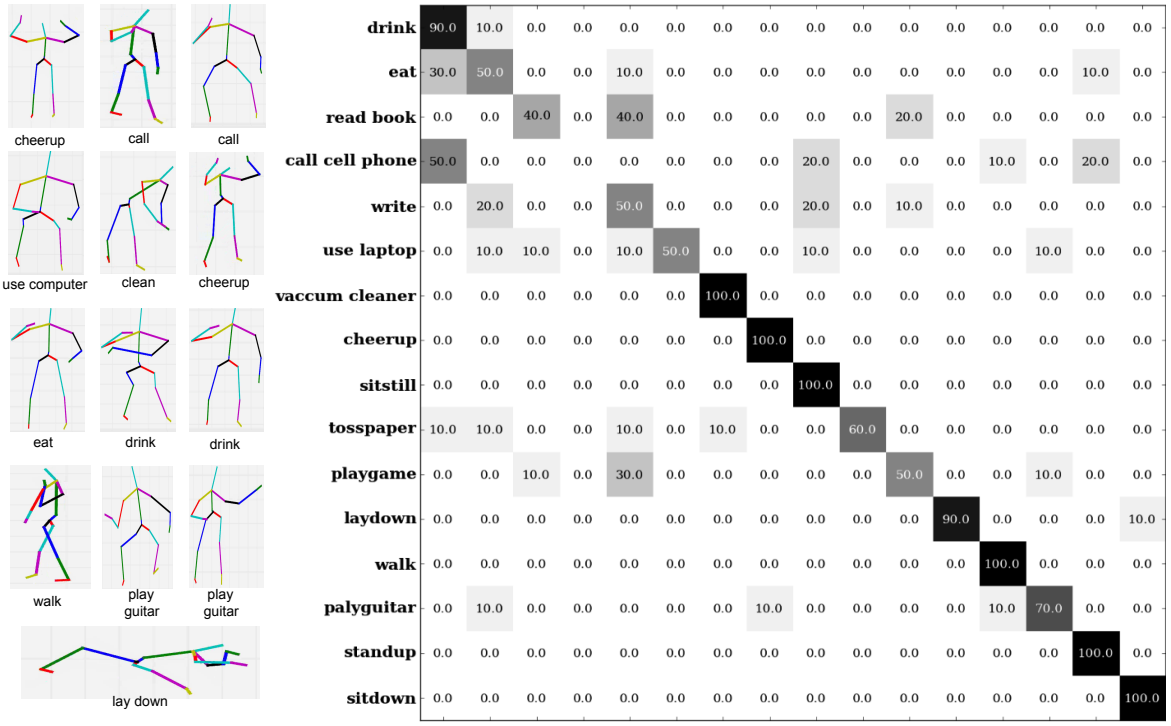


(a) Without temporal pyramid

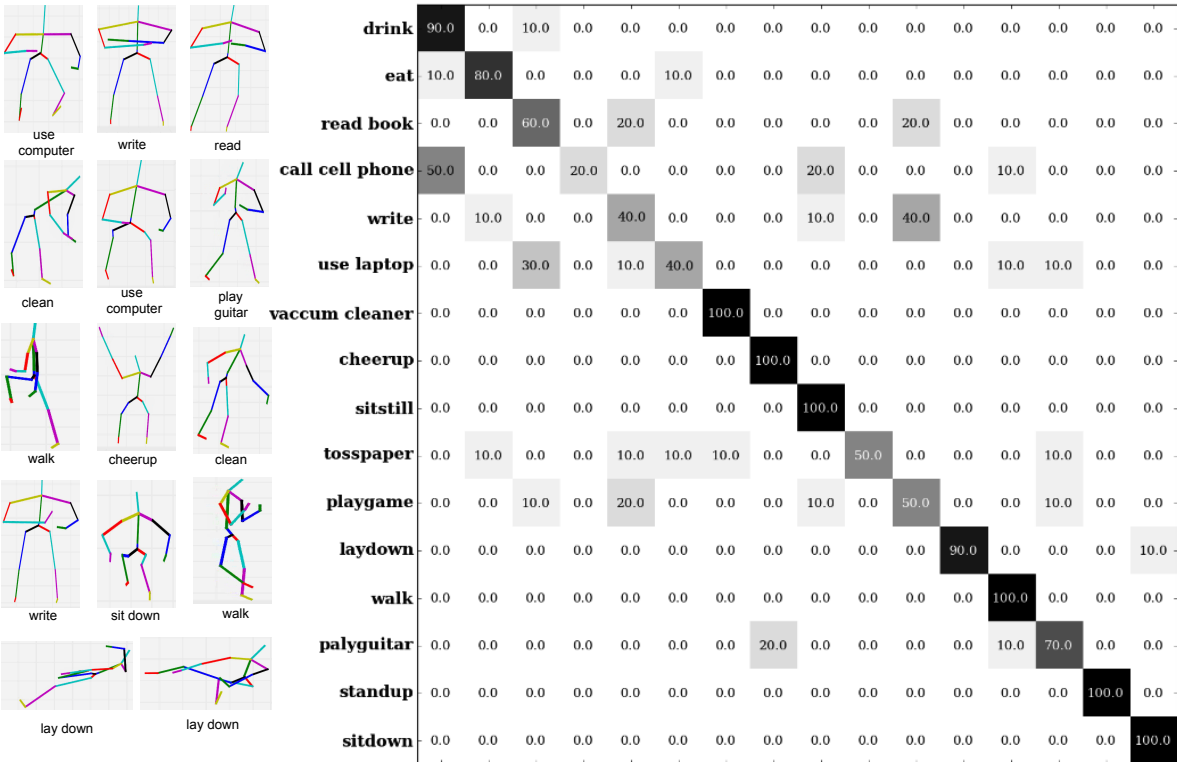


(b) With temporal pyramid

Figure 4.10: Confusion matrices for *3D Action Pairs* obtained (a) without a temporal pyramid, and (b) with a temporal pyramid.

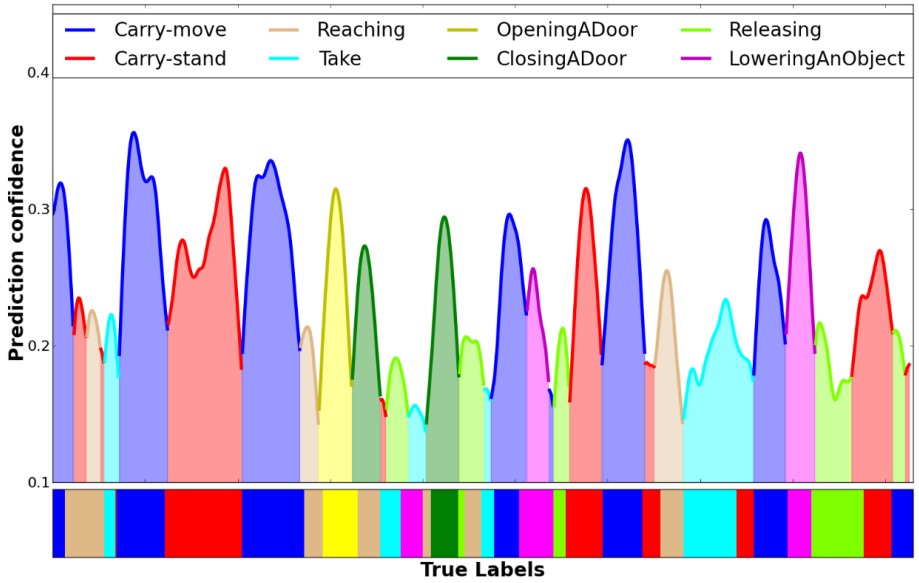


(a) Without temporal pyramid

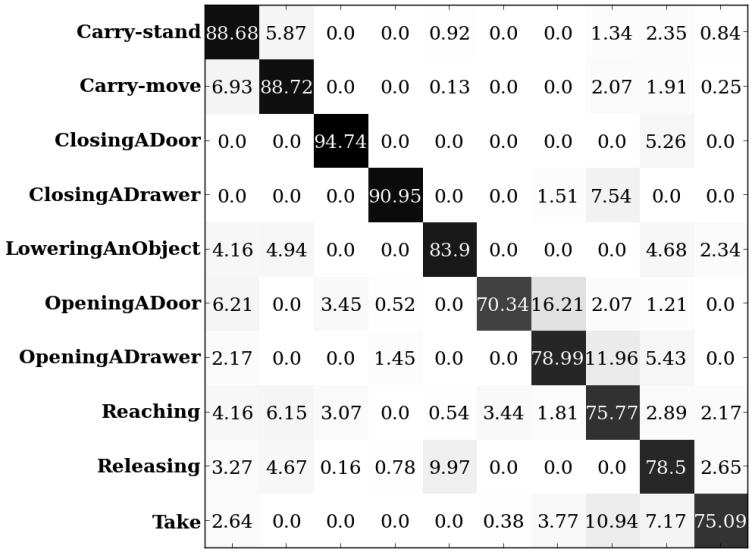


(b) With temporal pyramid

Figure 4.11: Confusion matrices for *MSR-DailyActivity* obtained (a) without a temporal pyramid, and (b) with a temporal pyramid.



(a)



(b)

Figure 4.12: Evaluation results on the *TUM Kitchen* dataset (a) Sample frame-level prediction where the x-axis shows the time span of the video sequence with ground-truth annotations and the y-axis shows classification accuracy. (b) Confusion matrix of the unsegmented video sequence of the *TUM Kitchen* dataset (best viewed in colors)

4.5 Summary

In this chapter, We have presented a novel framework for action recognition using high level pose representation depicted as 3D body poses. We have demonstrated the efficiency of this approach in terms of classification accuracy, training, and testing. We have also shown that the presented approach achieves state-of-the-art performance results on several human action benchmarks using only high level pose representation. This has been achieved by focusing on the most essential information that characterizes human actions, namely the relative locations of the joints, the velocities of the joints, and the correlations between locations and velocities denoted as movement normals. By combining these features with a discriminative approach to learn suitable bases, we obtain an action framework that outperforms state-of-the-art and is more efficient for training and testing.

In the following chapter, we evaluate our classification framework using only 2D body pose features that we have discussed in Section 4.3.1. This evaluation is motivated by the recent advances in 2D pose estimation (Yand and Ramanan, 2011; Yao, Gall and Gool, 2012) alongside the strong cues provided in several studies (Yao et al., 2011a; Tran, Kakadiaris and Shah, 2011) on the significance of pose representation for action recognition. We further compare the resulting performances achieved using 2D and 3D poses, and propose additional measures to enhance the performance on both 2D and 3D pose representation. We finally present an optimization approach using integral histograms for online classification scenarios where real time performance of human action recognition system is important.

CHAPTER 5

Evaluating Pose-based Variants for Action Recognition

5.1 Preface

The recent advancements in depth camera sensors and pose estimation algorithms rekindled researchers' interest in the importance of pose-based representations for human action recognition. Roughly, three different pose variants are currently used to represent the human body in current state-of-the-art pose estimation algorithms. These poses vary from being 2D, as it is the case in monocular pose-estimation techniques, or 3D in motion capture data and multiview pose estimation algorithms, or 3D with joints' orientations, as it is the case for the pose-estimator provided for the *Kinect* sensor data. Frequently, action recognition algorithms focus on using state-of-the-art pose-estimation algorithms only based on a recovery accuracy measure and not on the resulting pose variant. This chapter addresses the performance difference when different pose variants of 2D, 3D, and 3D with joints' orientation are used for human action recognition. In particular, we point the recognition gap between 2D poses and their corresponding 3D poses. To bridge this gap, we propose to map the 2D poses to their corresponding 3D poses, then use our earlier described view invariant features (see Section 4.3.1) from 3D instead of 2D poses. Despite the reconstruction error introduced by the 2D to 3D mapping, our experiments show a performance boost using the reconstructed 3D in comparison to 2D poses. This indicates that 3D pose estimation instead of 2D pose

estimation from monocular videos has the potential to improve action recognition. Furthermore, we show that enriching the 3D pose with joints' orientations benefits the recognition performance, especially for categorization of human gestures and actions of large intra-class and small inter-classes variation.

5.2 Introduction

Several human action recognition approaches invested significant effort using 2D and 3D pose representations. These representations are typically obtained from motion capture systems and pose-estimation algorithms that are frequently used as a preliminary stage for automatic action recognition frameworks. Early studies on human actions (Johansson, 1975) relied on the 2D pose representation to analyze how the human visual system perceives the body motion. Later, with the recent technological advancement, several approaches (Campbell and Bobick, 1995; Bissacco et al., 2001) were proposed to automatically recognize human action from 3D poses that were readily available from motion capture systems. Despite their success on several motion-capture benchmarks, the applicability of these approaches are still bounded to the expensive setups of motion capture systems which presume accurate pose estimation and tracking measurements.

The introduction of the *Kinect* sensor encouraged the presentation of affordable solutions for human pose estimation. Consequently, new application domains have emerged for action recognition and therefore, several studies were proposed in this line (Zanfir, Leordeanu and Sminchisescu, 2013; Wang, Wang and Yuille, 2013; Wang et al., 2012b; Wang, Wang and Yuille, 2013). A recent survey on human motion analysis from depth data was presented in (Ye et al., 2013). While these approaches show promising performance in terms of recognition accuracy, their applicability is constrained to a few environments where actions can be performed within the limited sensor's distance range with a pose facing the camera sensor.

Therefore, research devoted considerable effort to obtain a generic solution to the pose estimation problem in a way that does not impose any restrictions on the working environment. A survey of current trends in pose estimation techniques is presented in (Escalera, Angulo and Gonzalez, 2014). Most prominent approaches for pose estimation use structure-based or regression-based methods to recover the 2D body poses. Structured-based approaches as in (Yang and Ramanan, 2011) model the human body by a set of string-like connected parts arranged in a deformable configuration. Other approaches use a classification or re-

gression based scheme to recover the 2D body pose from rich feature representations that can uniquely codify the body pose appearance (Agarwal and Triggs, 2006a; Shotton et al., 2011). Due to the complexity of this problem, most pose estimation techniques for unconstrained environments focus on the pose estimation problem and overlook the optimal pose representation needed for human action recognition. Therefore, most provided solutions, except a few (Agarwal and Triggs, 2006b), limit their pose estimation algorithms to 2D poses for simplicity and do not leverage any 3D pose priors that can be beneficial in reconstructing 3D pose representations.

In this chapter, we revisit the problem of human action recognition and pose estimation in order to closely differentiate between the utility of different pose variants for human action recognition. Namely, we compare the action recognition performance when 2D, 3D, and 3D with joints' orientations are provided by the pose estimation algorithms. Furthermore, we underline the benefit of 3D pose estimation as opposed to 2D in obtaining view invariant pose features. Our results conclude that 3D pose estimation instead of 2D pose estimation from monocular videos has the potential to improve action recognition. The results also point out to the significance of estimating joints' orientations in boosting the recognition performance on challenging human actions benchmarks.

This chapter is organized as follows: We first review the theoretical bases of partial least squares for regression and classification which we use for mapping the 2D poses to 3D poses, Section 5.3. Next, we present our comparison in Section 5.4 between the recognition performances using 2D and 3D poses for the *TUM* dataset. In Section 5.4.2, we show how we use the PLS regression algorithm to reconcile the recognition performance difference between 2D and 3D poses. Section 5.5 describes how augmenting joints orientation of the 3D poses enhances action classification, especially for complex gestures of small inter-class and large intra-class variations.

5.3 Theoretical Review: Partial Least Squares (PLS)

PLS has been recently adopted in computer vision for different applications including pose estimation and regression (Haj, Conzalez and Davis, 2012), image classification (Harada et al., 2011), pedestrian detection (Schwartz et al., 2009), and multi-view learning (Sharma and Jacobs, 2011). The PLS algorithm is an iterative process which discovers relations between two blocks of data given by $\boldsymbol{X} \in \mathbb{R}^{n \times N}$ and $\boldsymbol{Y} \in \mathbb{R}^{n \times G}$ by learning a set of (non-) linear

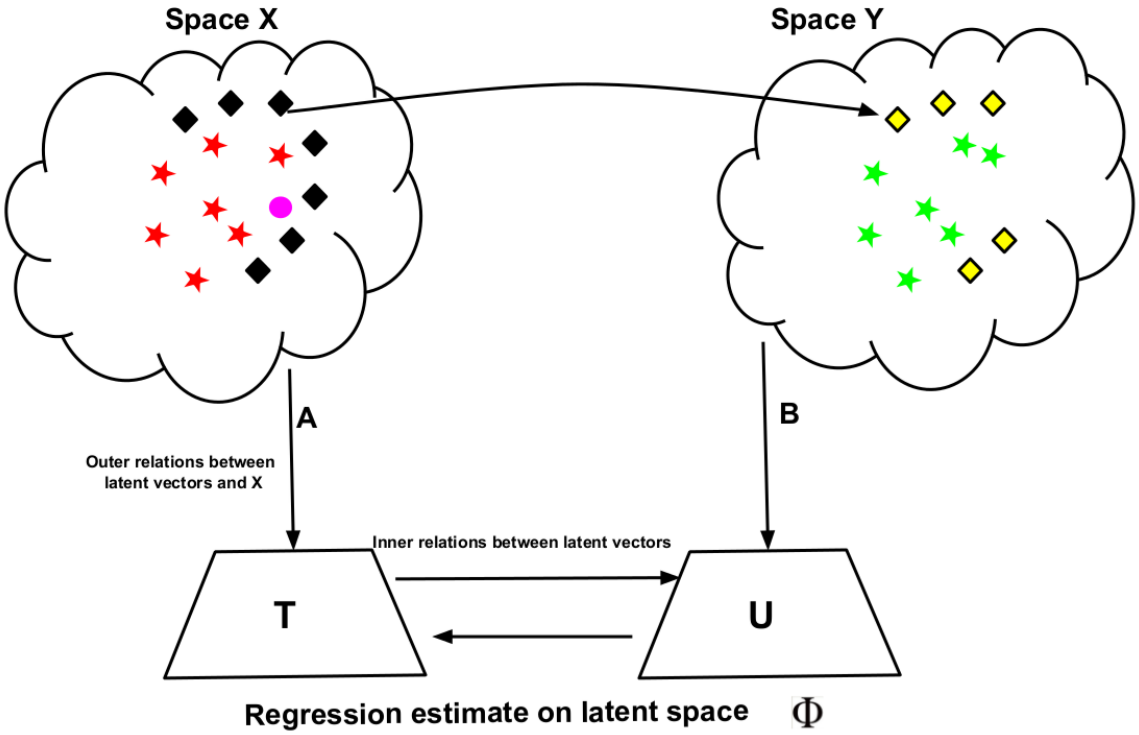


Figure 5.1: A schematic diagram that describes the PLS algorithm. The PLS algorithm discovers relations between two blocks of data by defining (non) linear outer relations between the input and the latent space, and inner relations between the variables in the latent space of the two blocks of data.

projections a, b (see Figure 5.1). This iterative process stands for two main subroutines: extraction of suitable weight vectors and deflation. Since its original introduction by Wold (1975), different forms have been proposed based on the followed deflation routine. As a complete review of PLS algorithms is beyond the scope of this thesis, we briefly describe here the PLS algorithm and its kernelized version. For a comprehensive review, we refer the reader to a detailed description and comparison between the PLS variants in (Wegelin, 2000).

5.3.1 The PLS Algorithm

Given zero-mean data $\boldsymbol{X} \in \mathbb{R}^{n \times N}$ and $\boldsymbol{Y} \in \mathbb{R}^{n \times G}$, PLS discovers relations between $\boldsymbol{X}$ and $\boldsymbol{Y}$ by learning a set of (non-) linear projections a, b onto p latent vectors that maximize covariance between both as follows:

$$\text{argmax} \left\{ \frac{[cov(a^T x, b^T y)]^2}{(a^t a)(b^T b)} \right\} \tag{5.1}$$

$$\begin{aligned} \boldsymbol{X} &= \boldsymbol{T}\boldsymbol{P}^{\boldsymbol{T}} + \boldsymbol{F} \\ \boldsymbol{Y} &= \boldsymbol{U}\boldsymbol{Q}^{\boldsymbol{T}} + \boldsymbol{H} \end{aligned} \tag{5.2}$$

where $\boldsymbol{T}$ and $\boldsymbol{U} \in \mathbb{R}^{n \times p}$ contain scores or latent vectors, and $\boldsymbol{P} \in \mathbb{R}^{N \times p}$ and $\boldsymbol{Q} \in \mathbb{R}^{G \times p}$ are loading matrices. PLS iteratively determines the p latent vectors $\boldsymbol{T}$ and $\boldsymbol{P}$ which, by design, are orthogonal vectors (Rosipal et al., 2001). The algorithm can be summarized as follows:

1. randomly initialize u_i

2. $a_i = \boldsymbol{X}^{\boldsymbol{T}} u_i, \quad t_i = \boldsymbol{X} a_i, \quad t_i = t_i / \|t_i\|_2$

3. $b_i = \boldsymbol{Y}^{\boldsymbol{T}} t_i, \quad u_i = \boldsymbol{Y} b_i \quad u_i = u_i / \|u_i\|_2$

4. repeat 2 and 3 until convergence

5. $p_i = \boldsymbol{X}^{\boldsymbol{T}} t_i / \|t_i\|_2$

6. $$\boldsymbol{X} = \boldsymbol{X} - t_i t_i^T \boldsymbol{X} \tag{5.3}$$

7. $$\boldsymbol{Y} = \boldsymbol{Y} - t_i t_i^T \boldsymbol{Y} \tag{5.4}$$

8. repeat p times.

9. Generate matrices by aggregating $[a_1, a_2, \ldots, a_p]$ into $A \in \mathbb{R}^{N \times p}$, $[b_1, b_2, \ldots, b_p]$ into $B \in \mathbb{R}^{G \times p}$, $[u_1, u_2, \ldots, u_p]$ into $U \in \mathbb{R}^{n \times p}$, and $[t_1, t_2, \ldots, t_p]$ into $T \in \mathbb{R}^{n \times p}$

Once the model parameters have been determined, the regression model for a new test instance amounts to

$$\boldsymbol{Y_t} = \boldsymbol{X_t}\boldsymbol{\Phi} \tag{5.5}$$

where $\mathbf{\Phi} \in \mathbb{R}^{N \times G}$, the coefficient matrix is given by $\mathbf{\Phi} = \boldsymbol{A}\ (\boldsymbol{P}^{\boldsymbol{T}}\boldsymbol{A})^{-1}\boldsymbol{B}^{\boldsymbol{T}}$ where $\boldsymbol{P} \in \mathbb{R}^{N \times p}$ is the matrix comprised of the loading vectors and the following holds:

$$\boldsymbol{A} = \boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{U} \tag{5.6}$$

$$\boldsymbol{P} = \boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{T}(\boldsymbol{T}^{\boldsymbol{T}}\boldsymbol{T})^{-1} \tag{5.7}$$

$$\boldsymbol{B} = \boldsymbol{Y}^{\boldsymbol{T}}\boldsymbol{T}(\boldsymbol{T}^{\boldsymbol{T}}\boldsymbol{T})^{-1}. \tag{5.8}$$

Similarly, the mapping from input features to their responses $\mathbf{\Phi}$ reads

$$\mathbf{\Phi} = \boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{U}(\boldsymbol{T}^{\boldsymbol{T}}\boldsymbol{X}\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{U}\boldsymbol{F})^{-1}\boldsymbol{T}^{\boldsymbol{T}}\boldsymbol{Y}. \tag{5.9}$$

This variant of PLS algorithm appears most frequently in chemometric literature and is often referred to as PLS1 or PLS2. If the number of dimensions G in $\boldsymbol{Y}$ is 1, its named PLS1, otherwise its named PLS2. Both are featured by the deflation routines followed by equation (5.3) and equation (5.4). Hence, both are updated by subtracting an estimate based on t, the latent variable score estimate for $\boldsymbol{X}$.

Other variants of PLS impose orthogonality constraints on the extracted weight vectors a and b (i.e. $a^T A = 0$), leading to different deflation routines than mentioned in equations (5.3) and (5.4). These correspond to the original deflation routine proposed by Wold (1975) (referred in literature as PLS-W2A) which subtracts rank-one estimates of the data matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$. After the updates, a new cross product of $\boldsymbol{XY}$ is estimated and the process iterates. Sampson, Streissguth and Bookstein (1989) demonstrate another variant of PLS called PLS-SVD which updates $\boldsymbol{XY}$ directly by rank-one minimization instead of updating $\boldsymbol{X}$ and $\boldsymbol{Y}$. In PLS-SVD the singular value decomposition (SVD) needs to be computed once on the original cross-product matrix. In contrast to PLS-W2A, the singular values must be estimated at each iteration after finding the new $\boldsymbol{XY}$ of the updated matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$.

5.3.2 The Kernel PLS Algorithm

Rosipal et al. (2001) introduce a kernelized extension for PLS so that it is possible to find non-linear transformations to the latent space through some mapping function $\mathbf{\Phi}(x_i)$. Using $\mathbf{\Phi}(\boldsymbol{x_i})\mathbf{\Phi}(\boldsymbol{x_i})^T = \boldsymbol{K}(\boldsymbol{x_i}, \boldsymbol{x_j})$, Rosipal et al. (2001) describe Kernel PLS as follows:

1. randomly initialize u_i

2. $t_i = \boldsymbol{K^T} u_i, \quad t_i = t_i / \|t_i\|$

3. $b_i = \boldsymbol{Y^T} t_i, \quad u_i = \boldsymbol{Y} b_i, \quad u_i = u_i / \|u_i\|$

4. repeat 2 and 3 until convergence

5. $\boldsymbol{K} = (\boldsymbol{I} - t_i t_i^T)\boldsymbol{K}(\boldsymbol{I} - t_i t_i^T)$

6. $\boldsymbol{Y} = \boldsymbol{Y} - t_i t_i^T \boldsymbol{Y}$

7. repeat p times

Similar to the original PLS algorithm, it is required to normalize both the kernel matrix $\boldsymbol{K}$ and the response matrix $\boldsymbol{Y}$ to have zero-mean. Finally, the predicted response matrix $\boldsymbol{Y_t t}$ can be obtained using regression coefficient as:

$$\boldsymbol{Y_t} = \boldsymbol{\Phi_t B} = \boldsymbol{K_t U}(\boldsymbol{T^T K U})^{-1} \boldsymbol{T^T Y} \tag{5.10}$$

where $\boldsymbol{K_t} \in \mathbb{R}^{(n_t \times n)}$) is the centralized kernel of n_t test samples (Schölkopf, Smola and Müller, 1998).

5.3.3 Partial Least Square for Classification

Partial least squares (Rosipal et al., 2001) is a regression method that models relationships between two sets of observed variables $\boldsymbol{X} \in \mathbb{R}^{n \times N}$ and $\boldsymbol{Y} \in \mathbb{R}^{n \times G}$ by projecting them to a common latent space where they are best aligned. Barker and Rayens (2003) point the utility of PLS for discrimination and established it relationship to Fisher Discriminant Analysis (FDA). The technique is known to be resistant to over-fitting, fast, easy to implement, and simple to tune. It often performs better than other regression approaches for classification, especially for *high dimension, low sample size data* (hall, Marron and Neeman, 2005). To accommodate the PLS algorithm for classification, Barker and Rayens (2003) propose using the label information of samples in $\boldsymbol{X}$ and encode it by the indicator matrix $\boldsymbol{Y}$ as:

$$\boldsymbol{Y} = \begin{bmatrix} \mathbf{1_{n_1}} & 0_{n_1} & \dots & 0_{n_1} \\ 0_{n_2} & \mathbf{1_{n_2}} & \dots & 0_{n_2} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{n_g} & 0_{n_g} & \dots & \mathbf{1_{n_g}} \end{bmatrix} \tag{5.11}$$

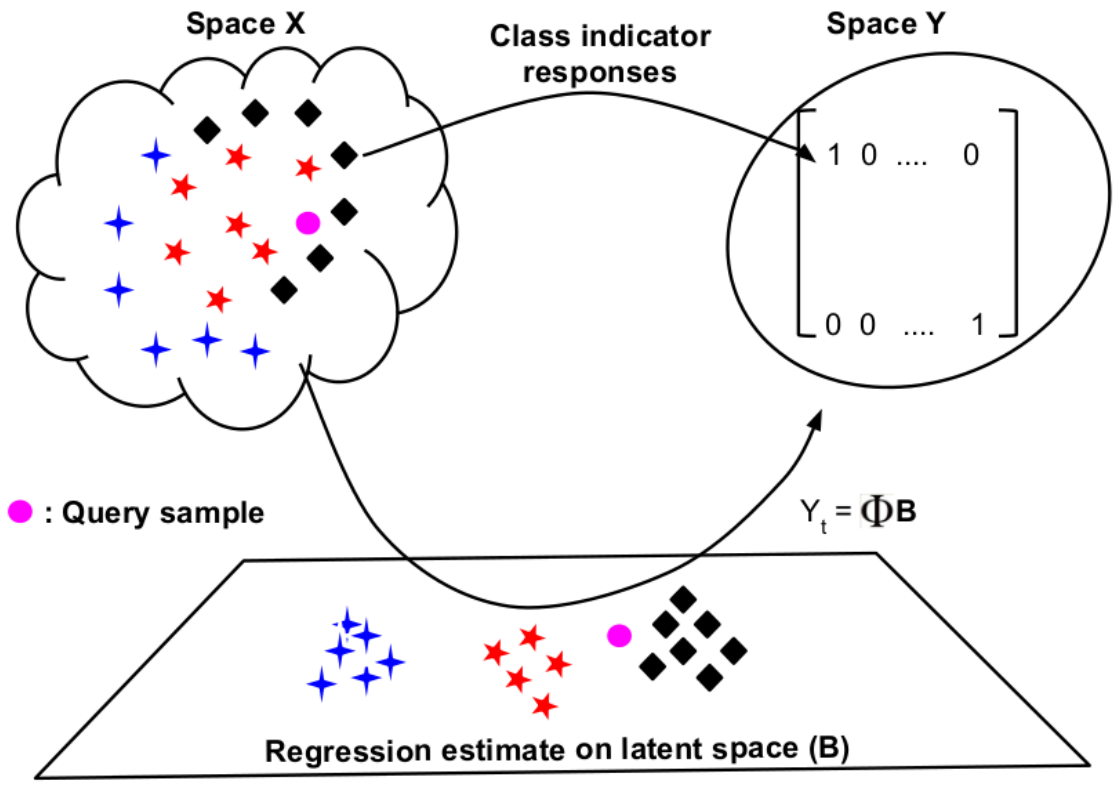


Figure 5.2: An overview of the classification approach using PLS, where the query sample is assigned to the class with the maximum regression score in the indicator vector Y_t.

where n_g denotes the number of instances of class g, see Figure 5.2. It was proven that using the indicator matrix as response for input features is equivalent to FDA (Barker and Rayens, 2003). However, FDA has the limitation that there are only $G - 1$ meaningful latent variables, when a G-class problem is considered. An advantage of PLS-type algorithms for classification is their efficiency and simplicity, even when applied to very high dimensional data. KPLS, as a special case, enables the use of kernel functions to deal with non-linearity and learn classification models that adapt better similarity measures between the samples for classification.

5.4 2D vs. 3D Pose Variants for Action Recognition

Recent advances in pose estimation introduce new opportunities towards action recognition in challenging environments. Several applications show that pose estimation is vital towards

generalizable, robust, and efficient action recognition (Yao, Gall and Gool, 2012; Tran, Kakadiaris and Shah, 2011; Jhuang et al., 2013). However, most estimation methods reconstruct 2D poses from monocular views, failing to provide view invariant descriptors for action recognition. In this section, we compare the performance of the pose features extracted from 2D and 3D pose variants for human action recognition. Our goal from this comparison is to present potential insights for pose estimation algorithms regarding the best pose variant needed for human action recognition.

5.4.1 2D Poses for Action Recognition

In the previous chapter, we have evaluated the performance of our action recognition framework for the *TUM* dataset (Tenorth, Bandouch and Beetz, 2009) using the estimated 3D poses. The *TUM* dataset (Section 2.2.2) is captured from four different calibrated views where the camera parameters are known. Therefore, it stands as a benchmark for comparing action recognition performance of 2D and 3D pose variants on the same action sequence. To this end, we utilize the provided camera parameters from the *TUM* dataset and obtain four different sequences of 2D human action representation; each depicts the human action from one camera view. Given the 3D action sequence from the *TUM* dataset of F frames for a human with J joints $S \in \mathcal{R}^{F\times J\times 3}$, and the four camera projection matrices of the *TUM* dataset $\{P_i\}_{i=1}^{4}$. We obtain the corresponding 2D action sequences $S_i \in \mathcal{R}^{F\times J\times 2}$ of each individual camera i by:

$$S_i = S\dot{P}$$

Afterwords, we perform action recognition using our earlier described framework, but only using the appropriate 2D features described in (Section 4.3.1). Namely, we use the velocity, and location features of the joints. Then, we perform soft-binning on the extracted orientation into 18 bins. Hence, the movement normal vectors can only be estimated as a scalar in the 2D case. It this can not be used for 2D sequences. The final features of velocity and location histograms are used to learn a classification model for each view. Similar to the training based on 3D pose presented earlier, we train the KPLS model for classification using the intersection kernel and follow the same evaluation protocol for the unsegmented sequences of the *TUM* dataset. The number of components used for the KPLS algorithms is learned by cross-validation on a separate portion of the training data. Table 5.2 summarizes the obtained classification accuracies for each view.

A common observation between results from 2D pose features is their relatively close recognition accuracies which are justifiable for the same action sequence. However, some views are slightly worse than others due to different degrees of view foreshortening. Notice also that the fourth camera have the worst recognition accuracy, which matches previous findings in (Yao, Gall and Gool, 2012) where they use a different set of features. Another observation is related to the classification models learned from 2D sequences, which provide a significantly worse performance of approximately 20% less than their 3D counterparts. This degraded performance can be justified by: Firstly, pose feature variation of the same 2D pose under different views; and secondly, the ambiguity of motion direction that is introduced when the depth information is absent. These observations motivate our proposal for 3D instead of 2D pose estimation from monocular views as they are expected to provide higher recognition performance. To confirm this view, the next section presents a method for reconciling the degraded performance of 2D action sequences by mapping them back to 3D, and comparing their recognition performance using the same framework.

5.4.2 3D Pose Mapping for Robust Action Recognition

In the previous section, we compared action recognition using 3D and 2D pose features. As opposed to the previously reported result for 3D pose features, action recognition accuracies in 2D show a significant drop of almost **20%** in recognition rates on all four cameras. The performance is also lower than the one reported for the 2D appearance features in (Yao, Gall and Gool, 2012), which does not use high-level features but low-level features based on optical flow and gradients.

In order to investigate if the performance loss comes from the inherent depth ambiguity of 2D poses or the view sensitiveness of the representation based on 2D poses, we propose to lift the 2D poses to 3D and reconstruct the depth information of the body pose. While learning-free approaches for pose-lifting (Taylor, 2000) provide pose mapping from 2D to 3D, their applicability to reconstruct a large sequence of 2D poses is limited, as they yield a whole set of 3D pose candidates for a single 2D pose (Brauer and Arens, 2011). Therefore, we resort to a regression-based approach which provides a one-to-one mapping from 2D to 3D poses. While any regression algorithm can be used to achieve this task, we chose the KPLS regression algorithm as it has shown state-of-the-art results in many head and body pose estimation applications (Haj, Conzalez and Davis, 2012; Sharma and Jacobs, 2011).

Given a set of 2D training poses of J joints $\{P\}_i^C \in \mathcal{R}^{2*J}$ and their corresponding 3D poses

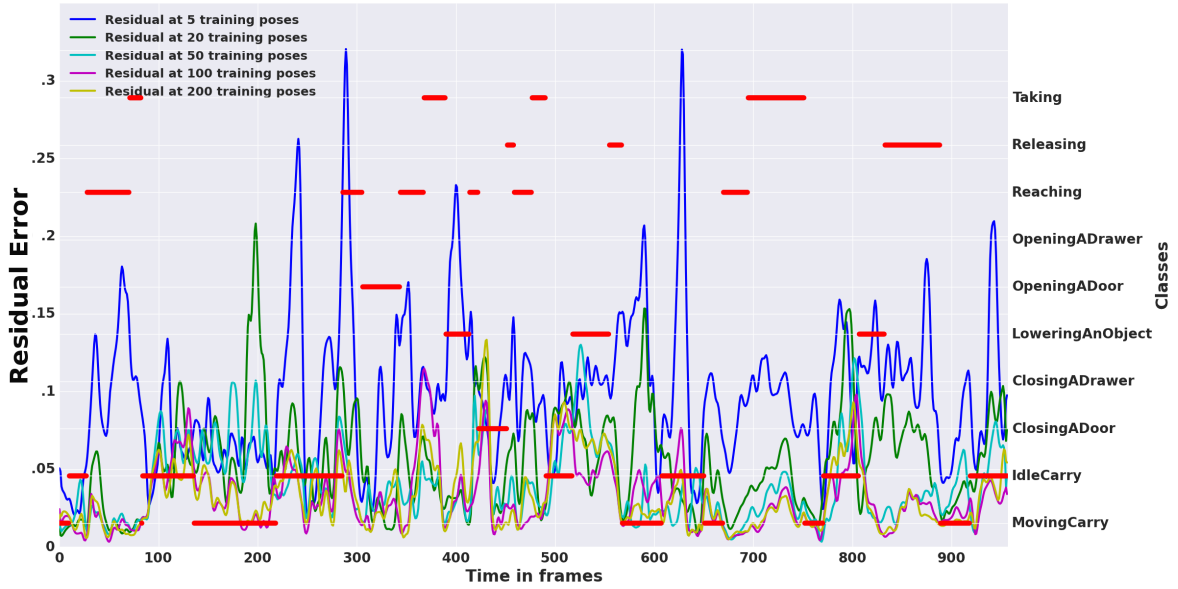


Figure 5.3: The residual error for each frame of a reconstructed 3D pose sequence when 5, 20, 50, 100, 200 poses per class are used to train the KPLS regression model.

$\{S\}_i^C \in \mathcal{R}^{3*J}$. We linearly scale the individual body parts so that the distance between the central shoulder and the central hip joints is constant. Then, we learn a mapping function $\Phi : \mathcal{R}^{2*J} \rightarrow \mathcal{R}^{3*J}$ that maps the observed 2D poses of a camera view to their 3D representation. As described earlier, we use the KPLS algorithm using a radial basis kernel with ($\sigma = 0.01$). Given a test sequence of 2D action sequence $S_i \in \mathcal{R}^{F\times J\times 2}$ captured from camera i, we learn its corresponding 3D pose sequence $S \in \mathcal{R}^{F\times J\times 3}$ using KPLS regression. Then, we use the reconstructed 3D pose sequence to extract the action features. For evaluation, we use the same protocol as described in Section 4.4.4 for the unsegmented sequences of the *TUM* dataset.

Figure 5.3 shows the residual error of the reconstructed 3D pose sequence compared with the corresponding original 3D pose sequence. Notice that using a smaller number of training poses leads to inaccurate reconstruction of the 3D body poses. Therefore, worse recognition performance is observed (Table 5.1) when compared to a larger number of training poses per action. Table 5.1 shows the recognition accuracy of our action recognition framework using the reconstructed 3D poses as a function of the number of training poses per action. Notice also that the recognition accuracy approximately converges to 78% when 100 or more poses are used for training the KPLS mapping.

Table 5.2 compares the obtained accuracies using our features after and before mapping

Table 5.1: Human action recognition using the reconstructed 3D poses when different number of poses per class is used to train the KPLS regression model.

Number of training poses per class	5	20	50	100	200
accuracy (%)	60.67	69.81	74.32	77.33	78.47

Table 5.2: Recognition accuracy (%) for the *TUM* dataset. We compare a 2D appearance-based approach (Yao, Gall and Gool, 2012), 2D versions of our features, 3D features obtained by mapping the 2D pose to 3D, and 3D features computed from the provided 3D poses, which have been estimated using all the four camera views.

Camera	Camera 1	Camera 2	Camera 3	Camera 4
HF + 2D appearance features (Yao, Gall and Gool, 2012)	68.00	70.00	68.00	65.00
KPLS + joint features from 2D pose	65.66	65.19	63.95	62.51
KPLS + joint features from 3D pose estimated from 2D pose of one camera view	77.61	77.78	78.23	78.47
KPLS + joint features from 3D pose estimated from all camera views	82.5			

the 2D poses to 3D. Despite the inaccurate reconstruction of the learned KPLS model, the corresponding 3D features show a significant performance boost over their 2D counterparts. It is also interesting to note that the performance is around 78% for all views, while the 2D features show more performance variation among views. Furthermore, the 2D appearance-based approach (Yao, Gall and Gool, 2012) is outperformed. This result underlines the benefit of view invariant pose features and indicates that 3D pose estimation instead of 2D pose estimation from monocular videos has the potential to improve action recognition.

5.5 Rich Pose-based Representation

We have described earlier how using 3D pose representation instead of 2D provided a significant potential in enhancing current monocular pose-based action recognition methods. In particular, we showed that this choice led to enhancing the action recognition performance by almost 15% on the *TUM* dataset. The new pose estimation algorithms nowadays accompanied with the *Kinect* sensor provides further information on the 3D poses to capture the 3D joints orientation. In this section, we briefly explain how to utilize this information and how it affects the performance of our action recognition framework (Chapter 4). For this purpose, we use the *ChaLearn* dataset which is captured using the *Kinect* camera sensor. The focus of

this dataset, as described in Section 2.2.2, is to recognize gestures drawn from a vocabulary of Italian sign gesture categories. The dataset emphasis is on user-independent, continuous gesture spotting of a set of 20 gestures performed by different users. It provides precise labels of gesture data that contains more than 900 samples, comprising near 14,000 gesture instances and more than 1.4 million frames. We first explain the provided joint information, and how we use it in our discriminative framework for action recognition. Then, we describe the use of temporal integral histogram for speeding the process of feature extraction. Finally, we elaborate on the evaluation setup for this dataset and empirically demonstrate the impact of using joints' orientations on the performance of our human action recognition system.

5.5.1 Joint Orientation in 3D Space

The 3D pose estimation algorithm accompanied with the *Kinect* provides rich representation of human pose that includes 3D joint location alongside their 3D orientations. Joint orientations are provided in terms of *unit quaternions* in 3D space. To obtain a directional orientation from the *unit quaternions* that expresses orientations of body's joints as a vector in 3D space, we convert the *unit quaternions* joint rotations x, y, z, w to the corresponding *rotation matrix*:

$$R = \begin{bmatrix} X_1 & Y_1 & Z_1 \\ X_2 & Y_2 & Z_2 \\ X_3 & Y_3 & Z_3 \end{bmatrix}$$

where:

$$X_1 = 1 - 2 \times y^2 - 2 \times z^2 \tag{5.12a}$$

$$X_2 = 2 \times x \times y + 2 \times z \times w \tag{5.12b}$$

$$X_3 = 2 \times x \times z - 2 \times y \times w \tag{5.12c}$$

$$Y_1 = 2 \times x \times y - 2 \times z \times w \tag{5.12d}$$

$$Y_2 = 1 - 2 \times x^2 - 2 \times z^2 \tag{5.12e}$$

$$Y_3 = 2 \times y \times z + 2 \times x \times w \tag{5.12f}$$

$$Z_1 = 2 \times x \times z + 2 \times y \times w \tag{5.12g}$$

$$Z_2 = 2 \times y \times z - 2 \times x \times w \tag{5.12h}$$

$$Z_3 = 1 - 2 \times x^2 - 2 \times y^2 \tag{5.12i}$$

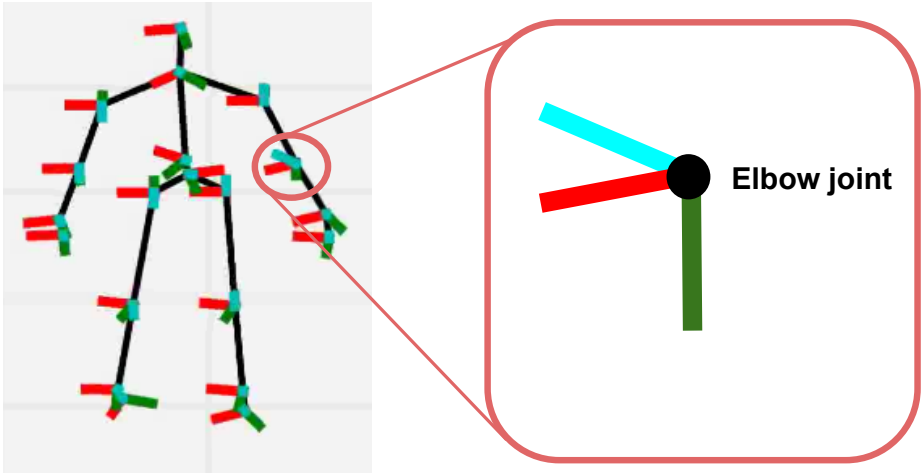


Figure 5.4: 3D pose representation with local joints orientation coordinates provided using the pose estimation algorithm accompanied with the *Kinect* sensor

The *rotation matrix* depicts the three local coordinates' orientation (X-Y-Z) of the body's joints on each of its columns (see Figure 5.4). Similar to the velocity and normal features described in Section 4.3.1, we extract the orientation feature by quantizing the polar coordinates' directions of azimuth (α) and zenith (ϕ) of a particular body joint's rotation axis (we used the Y axis) vector into 18×9 bins. In addition to the features previously described in Section 4.3.1, we add the orientation feature and learn a suitable basis to obtain a compact feature of the action sequence which can be used with an off-the-shelf classifier.

5.5.2 Experimental Details

To evaluate the significance of joint orientation in conjunction with 3D poses for action recognition, we utilize the *ChaLearn* dataset [1]. The evaluation scheme on this gesture dataset follows the original evaluation scheme introduced in the *ChaLearn 2014* gesture challenge (Section 2.2.2). The recognition performance of a test sequence is evaluated using the *Jaccard Index*, which provides a measurement of the overlap between the ground truth and the predicted labels of the video sequence. For each labeled gesture category from the 20 gestures, the *Jaccard Index* is computed as follows:

$$J_{s,n} = \frac{A_{s,n} \cap B_{s,n}}{A_{s,n} \cup B_{s,n}} \tag{5.13}$$

[1] gesture.chalearn.org/

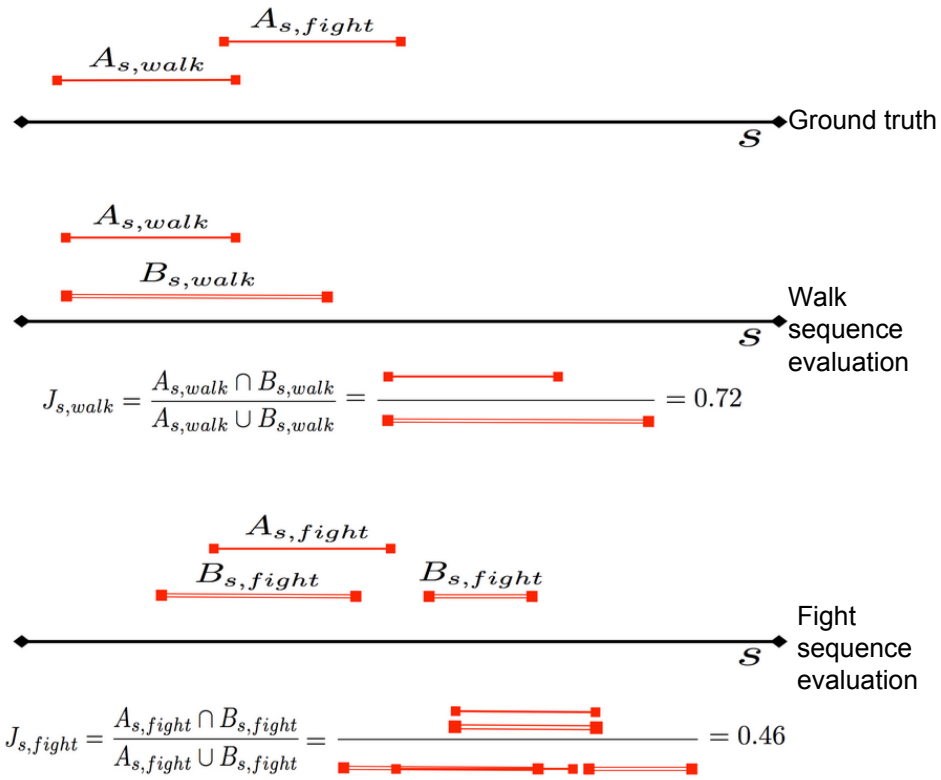
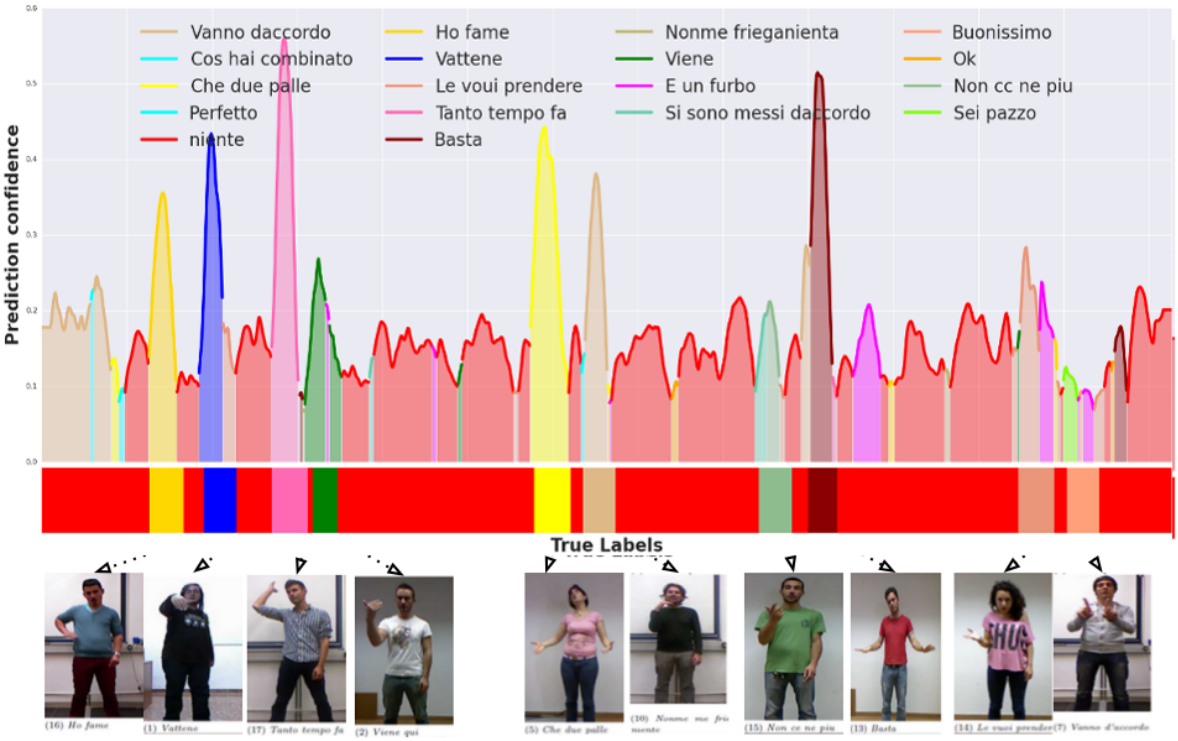


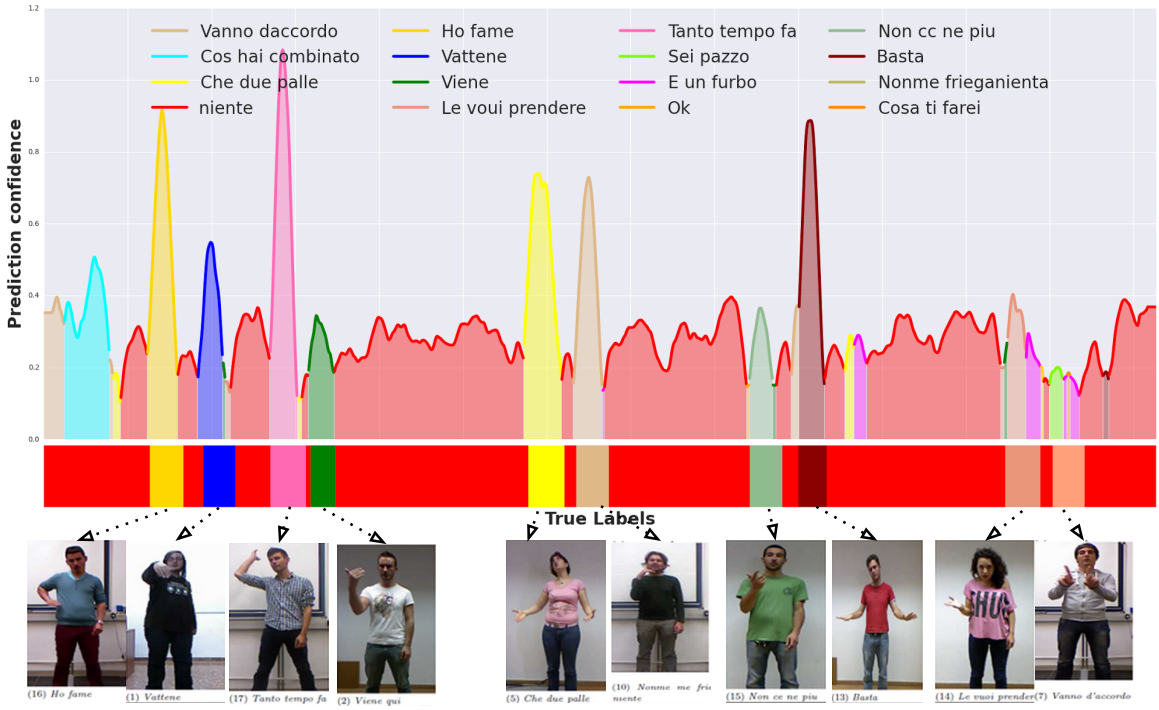
Figure 5.5: Performance evaluation of a two-class problem using the *Jaccard Index*. $A_{s,n}$ and $B_{s,n}$ denote the ground truth and prediction labels of sample s for each action respectively. The mean *Jaccard Index* is estimated by $J_{mean} = \frac{J_{s,walk}+J_{s,fight}}{2}$ which equals $J_{mean} = 0.59$

where $A_{s,n}$ and $B_{s,n}$ denote the ground truth and prediction labels of sample s for the gesture n respectively. Introducing false positives is yet penalized in the evaluation as the *Jaccard Index* of the class with false prediction is set to zero. Figure 5.5 shows an example evaluation of mean *Jaccard Index* for two action sequences, "walk" and "fight". The mean *Jaccard Index* is estimated by $J_{mean} = \frac{J_{s,walk}+J_{s,fight}}{2}$ which equals $J_{mean} = 0.59$. Accordingly, if the prediction present any false positive for an action that is not present in the ground truth, e.g."run", the mean mean *Jaccard Index* will be $J_{mean} = \frac{J_{s,"walk"}+J_{s,"run"}+J_{s,"fight"}}{3}$ which equals $J_{mean} = \frac{0.72+0+0.46}{3} = 0.39$ instead of 0.59.

Our implementation follows the same procedure earlier used on the *TUM* dataset. However, we introduce small changes to adapt the challenge of unlabeled and noisy motion that exists in this dataset, and limit the number of false positives that are strictly penalized by the performance measure. To train our classification model, we use a multiple-scale sliding window approach of 20, 30, and 40 frames around a labeled frame and extract training features from each temporal window. The use of multiple scales enriches the feature space,



(a) Prediction without joints orientation features



(b) Prediction with joints orientation features

Figure 5.6: Illustration of prediction confidence obtained from our classification framework for a sample video sequence that contains arbitrary Italian gestures. Figure (a) presents the prediction confidence obtained when joints orientation features are not used while (b) presents the prediction confidence when joints orientation is used in our classification framework.

especially at the transition points of different gestures and provides better training models for classification. We also run the sliding window approach over the unlabeled portion of the video sequence to better identify noisy movements and unlabeled gestures from known gestures. Notice that this process results in a large training data. Consequently, kernel-based classifiers like KPLS are impractical for this setup. Alternatively, we resort to the linear variant of PLS to train our classification model. We select the model hyper-parameters of the trained PLS classifier by cross-validation on a subset of the training data.

For testing, we predict action labels on a frame level basis using a sliding window approach. To speed up the computation of our features for a given window, we performed the quantization and soft-binning of the feature vectors (location, velocity, and movement normals) of the whole video sequence once. Then we compute an integral histogram representation along the temporal dimension. This reduces the computation of the feature vector of a given window to a constant time and brings the prediction time performance to real-time speed. The prediction can also be taken at multiple sliding window scales in order to cover actions with variant time lengths; this results in higher recognition performance and reduces the number of false positives introduced in the video sequence.

Our evaluation setup uses the provided 400/230 split as train/test samples. For model selection, we split the training data into 300 samples for training and 100 samples for validation. After setting the model parameters using the validation set, we retrain our classifier using the entire training data with the best learned parameters. Figure 5.6 compares the prediction confidence of a video sequence using our recognition framework with and without the orientation feature. Notice that the use of joints orientation feature improve the prediction performance in two ways: Firstly, it limits the number of false positives introduced by the model that is only trained using the 3D pose features. Secondly, it provids better discrimination between gestures that comprise similar motion but different hand poses such as the gestures of “Viene que” and “Tanto tempo fa”. The mean *Jaccard Index* on this dataset was 44.37% when only using the joints’ location, velocity, and movements normals. The orientation of the body’s joints features alone scored a mean *Jaccard Index* of 39.56%. This is a significant recognition rate when compared to the performance of the 3D joint locations. By combining both features in our framework, the mean *Jaccard Index* reaches up to 55.07%.

5.6 Summary

This chapter investigated different pose variants provided by state-of-the-art pose estimation algorithms. Our study focused on the utility of these pose variants and compared their performance for action recognition using the recognition framework presented in Chapter 4. In particular, our evaluation addressed the recognition gap between 2D and 3D poses for action recognition, and showed that 3D instead of 2D pose estimation from monocular videos has the potential to improve the recognition performance. Furthermore, we demonstrated how enriching the 3D pose representation with joints' orientations introduces substantial boost to the recognition performance, especially for the categorization of human sign gestures. These findings may motivate future generic pose estimation methods from monocular views to account for better pose variants in their formulation, and also to introduce better human action recognition frameworks.

CHAPTER 6

Optimal Late Fusion for Robust Action Recognition

6.1 Preface

Human action recognition in videos and still images has attracted considerable interest in recent research. A popular trend is to use ensembles of multiple features and classifiers in order to cope with different action's aspects such as action appearance, motion, and body pose. Most baseline approaches for ensemble learning combine a variety of complementary representations by simply combining their features or their corresponding confidence scores in a process that may undermine the discriminative potential of each individual representation for particular classes.

Motivated by the recent advances in ensemble learning techniques, especially for late fusion, we present in this chapter a novel framework for fusing the probabilistic predictions of different classifiers. Our approach is based on formulating and solving a constrained quadratic optimization problem. In contrast to the previously proposed late fusion approaches such as the sum-rule or linear weighting, our approach puts constraints on the semantics of mixture coefficients such that they represent the posterior of every participating classifier for each class. Unlike Bayesian inference methods, the proposed approach minimizes an error function that also considers correlations among different models. Experiments on a number of established benchmark action datasets show that the presented approach improves on

baseline late-fusion approaches and improves on state-of-the-art results.

6.2 Introduction

The problem of recognizing human activities from realistic images or videos has received considerable interest over the last decade. Accordingly, existing research has achieved promising advances in terms of informative features and efficient classification models. Despite this progress, human action recognition in unconstrained scenarios is largely an unsolved problem – mainly due to the challenges arising from the variation of human motion, appearance, scene, and body poses. To handle such variations, a practical system must incorporate representations based on a range of these cues. Most existing methods for action recognition typically rely on individual representations based on motion (Sadanand and Corso, 2012), pose (Thurau and Hlavac, 2008) or appearance features (Kuehne et al., 2011; Deltaire, Laptev and Sivic, 2010). Recently, Wang et al. 2013b obtained state-of-the-art results on several benchmark video datasets by encoding both motion and appearance features in a *BOF* model, affirming the benefits of combining several action representations.

The tangible performance enhancement achieved by using multiple features motivated several other attempts to combine various action representations. Frequently, these approaches rely on feature level fusion to achieve robust recognition. For instance, Deltaire, Laptev and Sivic (2010); Rohrbach et al. (2012); Wang et al. (2011a) combine a variety of heterogeneous representation by simply concatenating feature descriptors. This, however, may undermine the discriminative potential of each individual representation for particular classes. To overcome this limitation, Wang et al. (2012a) follow a principled approach to combine a set of mined action features called *actionlets* using Multiple Kernel Learning (MKL) (Bach, Lanckriet and Jordan, 2004). MKL assigns different linear or non-linear weights to the feature kernels in order to obtain better similarity measures for the purpose of classification. Yet, a recent evaluation Gehler and Nowozin (2009) show that simple kernel averaging, a much faster method, can achieve similar results as MKL.

Classifier level fusion, often called late fusion, has been widely used and baselines methods have been thoroughly investigated (Kittler et al., 1998; Xu, Krzyzak and Suen, 1992). Researchers observed that performing classifier level fusion has certain key advantages over other fusion schemes. Firstly, classifier level fusion is generally faster than feature level schemes, especially as the trained system grows to adapt new features. In this case, classifier

level fusion requires only to re-train the fusion part in contrast to feature level fusion where the whole system needs to be retrained. Secondly, it abstracts the details of the underlying classifiers, giving the freedom of selecting classification models that best suit a given feature.

Baseline approaches for classifier level fusion such as the sum-rule or the SVM-rule (Kittler et al., 1998) have been extensively evaluated for several applications (Kittler et al., 1998; Xu, Krzyzak and Suen, 1992). These baselines assume that individual classifier outputs are normalized to an estimate of posterior probabilities so that they can be combined homogeneously (Jain, Duin and Mao, 2000). Despite their good performance, these approaches are frequently criticized as they neglect the discriminative power of features with respect to particular classes, thus leading to suboptimal fusion performance. To remedy this limitation, alternative methods have been suggested that learn weights for classifier scores (Terrades, Valveny and Tabbone, 2009), clustering results (Liu et al., 2012), or even on data samples (Liu et al., 2013a).

This chapter presents a novel late fusion strategy that determines stochastic weights of the *models* for each class. Our approach is based on a quadratic optimization formulation. Unlike common linear weighting schemes for late fusion, our approach constraints the semantics of the mixture coefficients (weights) in order to represent posteriors of a model for each class. We evaluate our fusion scheme on different human action datasets comprising videos and images. Our experimental results show that the proposed late fusion approach outperforms other late fusion techniques and provides state-of-the-art classification accuracies on various action recognition datasets.

6.3 Related Work

Fusing complementary modalities and feature representations became popular trend in computer vision research. Conventional approaches such as kernel averaging and the sum-rule have been widely adopted for their simplicity and ease of implementation (Kittler et al., 1998; Xu, Krzyzak and Suen, 1992). Alternatively, a principled early fusion strategy consists in *Multiple Kernel Learning* (MKL) (Bach, Lanckriet and Jordan, 2004) which aims at optimized combinations of kernels. For instance, He et al. (2008) formulate a quadratic optimization approach to learn optimal discriminative linear kernel weights for classification. However, Gehler and Nowozin (2009) extensively evaluate *MKL* and found that even baseline approaches such as kernel averaging can be as effective as *MKL*. In contrast to these

approaches, our efforts focus on late fusion which builds on the confidence scores of different models of different features.

Since late fusion techniques are based solely on the predictions obtained from different models, they provide greater flexibility w.r.t. the choice of models. Kittler et al. (1998) evaluate baseline strategies for late fusion and concluded that the sum-rule with uniform linear weighting performs best in almost all situations. Most existing linear weighting approaches for late fusion use equal (sum), static, or classifier level fusion weights (Atrey et al., 2010). Other approaches (Tavakoli, Zhang and Son, 2005; Atrey, Kankanhalli and Jain, 2006) use domain heuristics or histories of classifiers to assign weights to individual classifiers. However, adapting such naive approaches may severely affect the final results in situations where a specific model performs poorly on certain classes. Different attempts were made further to learn better late fusion schemes that identify non-discriminative models in order to ignore them in the final decision. Nandakumar et al. (2008) fit a Gaussian mixture model to the scores of different features and then utilized a likelihood ratio test to fuse classifier scores. Terrades, Valveny and Tabbone (2009) develop a late fusion approach that optimizes for the best linear combination in terms of the misclassification rate under L_1 constraints for multiple binary classifiers. In contrast to these approaches, our approach considers the individual results of each model for each particular class in an optimization setup and determines a joint stochastic linear weighting of individual models for each class.

Another late fusion scheme was recently presented in (Ye, Liu and Chang, 2012) where they develop a novel method for fusing results of multiple models via rank minimization on the pairwise relation matrices of the learned models. Consequently, their approach ignores model confidence scores which, however, are of great interest for indexing and retrieval. Recently, Liu et al. (2013a) present a promising approach that adopts a sample-specific late fusion scheme by propagating the learned fusion weights of labeled samples to unlabeled samples. Again, our approach differs from these ideas as we determine class level weights in a supervised fashion. Finally, for an extensive review of related approaches in multimedia retrieval, we refer the reader to a recent survey in (Atrey et al., 2010).

In the context of action recognition, fusion of multiple modalities is of particular interest since different actions are often best characterized in terms of different representations pertaining to motion, appearance, scene, and body pose. Research on combining these modalities, however, still lacks an exhaustive evaluation. Deltaire, Laptev and Sivic (2010); Wang et al. (2011a) utilize different features of spatial or spatio-temporal representations and com-

bine them by means of averaging their kernels. Wang et al. (2012a) learn a linear combination of mined actionlets for classification which may not bring a significant enhancement on the classification results compared to simple kernel averaging (Gehler and Nowozin, 2009). Below, we address these limitation and propose a stochastic late fusion technique based on quadratic optimization that jointly learns the best linear combination of models in a multi-class classification scenario. The learned weights can reveal the significance of the utilized features as well as the discriminating potential of each model for their respective classes.

6.4 Late Fusion: Baseline Approaches

Most baseline approaches for late fusion assume a Bayesian framework to justify the merits behind using a specific fusion scheme. In this section, we review the theoretical background of most popular approaches for late fusion in human action recognition and their relation to the Bayesian theory. Then we present our optimal late fusion formulation for human action recognition. Let's assume that the action sample S is to be assigned to one of the C classes $(\omega_1, \omega_2, \ldots, \omega_C)$, and that each sample S is represented through $\mathcal{M}$ different view representations which cast the measurements vectors $[x_1, x_2, \ldots, x_\mathcal{M}]$. In the feature space, we can model the probability density function of each class ω_c as $P(x_i|\omega_c)$ with a priori probability $P(\omega_c)$. In a Bayesian framework, the pattern S should be assigned to the class ω_j that maximizes the posterior probability of:

$$P(\omega_j|x_1, \ldots, x_\mathcal{M}) = \underset{c}{\operatorname{argmax}}\, P(\omega_c|x_1, \ldots, x_\mathcal{M}) \tag{6.1}$$

In principle, the computation of the posterior probability depends on estimating the joint probability density function $P(x_1, \ldots, x_\mathcal{M})$ which is difficult to compute. Therefore, most late fusion techniques relax this term by introducing further assumption to the problem formulation. Next, we present the baseline approaches for late fusion and describe the assumption taken in their formulation and their reasoning in the problem for human action recognition.

6.4.1 Product Rule

As the features obtained from different action representations $[x_1, x_2, \ldots, x_\mathcal{M}]$, it becomes convenient to assume their independence. This assumption can be reflected to the formulations as:

$$P(\omega_c|x_1,\ldots,x_{\mathcal{M}}) = \frac{P(\omega_c)\prod_{i=1}^{\mathcal{M}} P(x_i|\omega_c}{\sum_j^C P(\omega_j)\prod_{i=1}^{\mathcal{M}} P(x_i|\omega_j)} \tag{6.2}$$

Using both (6.1) and (6.2), we can assign the action sample S to the class ω_j that maximizes

$$P(\omega_j)\prod_{i=1}^{\mathcal{M}} P(x_i|\omega_j) = max_{c=1}^{C} P(\omega_k)\prod_{i=1}^{\mathcal{M}} P(x_i|\omega_k) \tag{6.3}$$

which can be written in terms of posterior probability as

$$P^{-(R-1)}(\omega_j)\prod_{i=1}^{\mathcal{M}} P(\omega_j|x_i) = max_{c=1}^{C} P^{-(R-1)}(\omega_c)\prod_{i=1}^{\mathcal{M}} P(\omega_c|x_i) \tag{6.4}$$

6.4.2 Sum Rule

While the product rule presumes the conditional independence among observations from various modalities, the sum rule can be yet understood with an additional strong presumption. According to Kittler et al. (1998), the sum rule assumes that the posterior probability of the observations slightly deviates from the prior probabilities. It therefore can be described as:

$$P(\omega_c|x_i) = P(\omega_c)(1+\delta_{ci}) \tag{6.5}$$

where $\delta_{ci} \ll 1$. Substituting the posteriori 6.5 in 6.4 we find

$$P^{-(R-1)}(\omega_c)\prod_{i=1}^{\mathcal{M}} P(\omega_k|x_i) = P(\omega_c)\prod_{i=1}^{\mathcal{M}}(1+\delta_{ci}) \tag{6.6}$$

By expanding the right-hand side and neglecting any second or higher order, we reach

$$P(\omega_c)\prod_{i=1}^{\mathcal{M}}(1+\delta_{ci}) = P(\omega_c)\sum_{i=1}^{\mathcal{M}}\delta_{ci} \tag{6.7}$$

Substituting 6.7 and 6.5 into 6.4 we obtain the sum decision rule

$$(1-R)P(\omega_j) + \sum_{i=1}^{\mathcal{M}} P(\omega_j|x_i) = max_{c=1}^{C}\left[(1-R)P(\omega_k) + \sum_{i=1}^{\mathcal{M}} P(\omega_c|x_i)\right] \tag{6.8}$$

6.4.3 Bayesian Inference Rule

Baseline methods for late fusion (i.e. product- and sum-rule) are often criticized because they undermine the prior knowledge of individual performance of each modality in discrimination, especially when the modalities provide variant accuracy performances across the classes. Such situations are particularly frequent in the field of action recognition where for example, motion features are very effective for detecting motion-based actions (e.g. running) and poor in static-based action (e.g. reading). Therefore, it is crucial to allow for any prior knowledge about the likelihood of the modality for a particular action to be utilized in the inference process. In short, the Bayesian inference approach can be described by

$$P(c|x_1, \ldots, x_{\mathcal{M}}) = \underset{c}{\operatorname{argmax}} \sum_{i=1}^{\mathcal{M}} P(c|m, x_i) P(c|m_i) P(m_i). \tag{6.9}$$

where $P(c|m, x_i)$ and $P(m_i)$ are the prior parameters of the modality m_i and the modality m_i for a given class c. However, these methods may not introduce much enhancement on the fusion performance because of the absence of knowledge of suitable priors. Our presented approach addresses this flow in Bayesian inference approaches by learning suitable priors for the action modality through an optimal late fusion formulation that we describe next.

6.5 Late Fusion by Quadratic Programming

In the previous section, we have reviewed baseline approaches for late fusion. We have pointed out that both sum- and product-rule are widely criticized as they do not allow any prior knowledge of the action modalities into fusion. Furthermore, we have discussed that the Bayesian inference approach for late fusion may not be suitable because of the absence of knowledge of suitable priors. In this section, we present an optimal late fusion approach that utilizes the likelihood prior of each action modality into the fusion by learning an optimal fusion weights using a quadratic optimization approach. This section describes our optimization approach to late fusion and its application for human action recognition.

Let $\mathbf{D}$, $\mathbf{V}$, and $\mathbf{T}$ be three independent sets of data and let M be the number of constituting models trained on a the training set $\mathbf{D}$ which contains C classes or categories. Further, let N be the number of samples in the validation set $\mathbf{V}$ that will be used for learning the fusion model. Given that each model provides a probabilistic predictions, let $\mathbf{V}^{(m)}$ be an $N \times C$

matrix of predictions of all the N instances according to model m, i.e. each row of $\mathbf{V}^{(m)}$ is a stochastic vector. Let $\mathbf{Y}$ be an $N \times C$ binary indicator matrix based on true labels such that $y_{ic} = 1$ only if sample i belongs to category c. Then, our objective is to find stochastic mixture coefficients for each class and each model that minimize the sum of squared errors over all in the training data. Let $\mathbf{w}_m$ denote the C-dimensional column vector of the target mixture coefficients for model m and let $\odot$ represents the Hadamard (element-wise) product of each row of a matrix with a row vector, then our objective is to solve

$$\begin{aligned} \min_{\mathbf{w}_m} \quad & \left\| \mathbf{Y} - \sum_m \mathbf{w}_m^T \odot \mathbf{V}^{(m)} \right\|_F \\ \text{s.t.} \quad & \sum_{m=1}^{M} w_{mc} = 1, \quad w_{mc} \geq 0 \quad \forall m, c \end{aligned} \tag{6.10}$$

where $\|.\|_F$ denotes the Frobenius norm. Solving this system yields weights w_{mc} that encode the belief of a model m regarding its performance for class c. The sum-to-one constraint in the above formulation ensures that weights of different models are normalized for each class and hence that beliefs are measured relative to each class.

In the remainder of this section, we discuss further details regarding the formulation of the above quadratic optimization problem and present a strategy for its solution. Note first that the objective function in (6.10) is equivalent to

$$\min_{w_{mc}} \sum_{i=1}^{N} \sum_{c=1}^{C} \left(y_{ic} - \sum_{m=1}^{M} w_{mc} v_{ic}^{(m)} \right)^2 \tag{6.11}$$

where $v_{ic}^{(m)}$ represents the probability that sample i belongs to class c according to model m. Expanding this expression yields the following coefficients of the unknowns

$$coeff\left(w_{mc}^2\right) = \sum_{i=1}^{N} \left(v_{ic}^{(m)}\right)^2 \tag{6.12}$$

$$coeff\left(w_{mc} w_{kc}\right) = \frac{1}{2} \sum_{i=1}^{N} v_{ic}^{(m)} v_{ic}^{(k)} \tag{6.13}$$

$$coeff\left(w_{mc}\right) = -2 \sum_{i=1}^{N} y_{ic} v_{ic}^{(m)} \tag{6.14}$$

Next let $\mathbf{w}$ be a MC-dimensional column vector obtained by stacking the $\mathbf{w}_m$ such that

$$\mathbf{w} = [w_{11}w_{12}\dots w_{1C}w_{21}w_{22}\dots w_{MC}]^T . \tag{6.15}$$

Further consider $\mathbf{P}$ to be a $MC \times MC$ matrix whose $C \times C$ blocks contain coefficients corresponding to the quadratic terms in (6.11). Specifically, $\mathbf{P}$ has the following shape:

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{1,1} & \mathbf{P}_{1,2} & \cdots & \mathbf{P}_{1,M} \\ \mathbf{P}_{2,1} & \mathbf{P}_{2,2} & \cdots & \mathbf{P}_{2,M} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{M,1} & \mathbf{P}_{M,2} & \cdots & \mathbf{P}_{M,M} \end{bmatrix} \tag{6.16}$$

where each $\mathbf{P}_{m,k}$ is $C \times C$ diagonal (sub)matrix and contains the coefficients $coeff\left(w_{mk}^2\right)$ if $m = k$ and the coefficients $coeff\left(w_{mc}w_{kc}\right)$ otherwise. Note that $\mathbf{P}$ is positive (semi)definite as it contains blocks of positive (semi)definite matrices.

Also let $\mathbf{q}$ be a $1 \times MC$ stacked vector containing coefficients of the linear terms in (6.11). That is

$$\mathbf{q} = [\mathbf{q}_1\mathbf{q}_2\cdots\mathbf{q}_M] \tag{6.17}$$

where each $\mathbf{q}_i$ is a $C-$dimensional row vector.

Accordingly, the problem defined in (6.10) is equivalent to the following standard quadratic program

$$\begin{aligned} \min_{\mathbf{w}} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{P}\mathbf{w} + \mathbf{q}\mathbf{w} \\ \text{s.t.} \quad & \mathbf{I}_{C\times MC}\mathbf{w} = \mathbf{1} \\ & \mathbf{I}_{MC\times MC}\mathbf{w} \geq \mathbf{0} \end{aligned} \tag{6.18}$$

where $\mathbf{I}_{C\times MC}$ is a matrix containing stacked identity matrices of dimension C and $\mathbf{0}$ and $\mathbf{1}$ are MC dimensional column vectors containing zeros and ones, respectively.

The optimal solution of this convex problem determines the mixture coefficients w_{mc} such that each such coefficient can be interpreted as $P(c|m)$. A given query instance x is then

classified accordingly, i.e.

$$\underset{c}{\operatorname{argmax}}\, P(c|x) = \underset{c}{\operatorname{argmax}} \sum_{m=1}^{M} P(c|m,x)P(c|m)P(m). \tag{6.19}$$

where $P(c|m,x)$ is the probabilistic prediction by model m, $P(c|m)$ is the stochastic weight learned by our approach and $P(m)$ is the prior probability of the model. The prior $P(m)$ can be considered uniform or can be estimated in terms of average accuracies through cross-validation in the training/validation phase (Atrey, Kankanhalli and Jain, 2006).

6.5.1 Regularization and Normalization

Formulating the energy function for optimization problems such as the one in (6.10) allows us to penalize the weight vector and to reduce effects due to unbalanced data.

Handling unbalanced data: Unbalanced data may cause a bias in the objective function in (6.11). This, however, is easily overcome by reformulating the objective function as

$$\min_{w_{mc}} \sum_{i=1}^{N} \sum_{c=1}^{C} \frac{1}{\sum_{i=1}^{N} y_{ic}} \left(y_{ic} - \sum_{m=1}^{M} w_{mc} v_{ic}^{(m)} \right)^2. \tag{6.20}$$

This formulation normalizes the artifacts of having different sizes for different categories in the training data. Readers may verify that this formulation affects only the diagonal entries in the $\mathbf{P}$ matrix in (6.16).

Regularizing and penalizing the weight vector: The energy function formulation permits further parametrization to introduce certain properties of optimal solution, e.g. sparsity (L_1-normalization) or smoothness (L_2-normalization). Note that L_1 regularization is embedded in our framework as a constraint, i.e. weight vectors must be stochastic. The L_2 regularization can be added to the objective function which will become

$$\min_{w_{mc}} \sum_{i=1}^{N} \sum_{c=1}^{C} \frac{1}{\sum_{i=1}^{N} y_{ic}} \left(y_{ic} - \sum_{m=1}^{M} w_{mc} v_{ic}^{(m)} \right)^2 + \lambda \sum_{c,m} w_{mc}^2 \tag{6.21}$$

where λ is a regularization constant and can be evaluated through cross-validation.

6.6 Datasets and Feature Descriptors

In this section, we briefly discuss three well known action datasets containing challenging videos and images and provide details as which features are extracted from each dataset.

6.6.1 HMDB Video Dataset

HMDB (Kuehne et al., 2011) is one of the largest and most versatile datasets for action recognition in videos. It contains 6,766 video sequences of 51 action categories such as facial actions, body movements, and human interactions. In our experiments on this dataset, we considered the following feature descriptors which are known to show good performance.

Action Bank (Sadanand and Corso, 2012)

Action bank consists of a set of high level action detectors sampled broadly in semantic space and viewpoint spaces. Action bank feature extraction is based on spotting different motion templates in the multiple scale spatio-temporal cuboids. We used the same settings as in (Sadanand and Corso, 2012) to extract $14,965$ dimensional features. They have shown good performance in combination with linear SVM classification.

HOG/HOF Around Harris3d Corners (Laptev et al., 2008)

Histogram of oriented gradient (*HOG*) and histogram of oriented flow (HOF) features are determined around space time interest points (e.g., Harris 3D corners) are often considered as baseline. We used the binaries provided by Laptev et al. (2008) to extract *HOG/HOF* features along STIPs. A Bag-of-Features (*BOF*) method is adopted by first sampling 100,000 descriptors from the training data and then building a vocabulary of 2000 words using k-means clustering. For a video, each descriptor is quantified to the nearest word in the vocabulary and the resulting histograms are normalized to have unit sum. The best baseline classifier is an SVM with a Gaussian kernel.

Motion Boundary Histograms and HOG/HOF Along Dense Trajectories (Wang et al., 2011a)

Since most STIP detectors (e.g. Harris3D) are extensions of their 2D counterparts, they may fail to identify or keep track of interesting spatio-temporal regions. To this end, (Wang et

Table 6.1: Recognition accuracies (%) of different approaches

Model		Datasets		
		HMDB	*Web-actions*	*PPMI7*
Individual	**Action Bank**	25.90	-	-
	STIP *HOG/HOF*	18.45	-	-
	Dense Trajectories MBH	34.31	-	-
	Dense Trajectories *HOG/HOF*	30.13	-	-
	HOG	-	57.4	68.7
	***BOF*-Sift**	-	56.8	63.6
Late Fusion	**Bayes**	40.76	65.6	72.4
	Sum Rule	40.70	66.5	72.9
	SVM	40.63	64.2	70.32
	Our Approach	**41.83**	**67.15**	**74.0**

al., 2011a) proposed an efficient way to track densely sampled sampled points using optical flow fields. They also proposed a novel feature descriptor based on motion boundary histograms which is robust to camera motion. Wang et al. (2013b) use multiple features along dense trajectories and achieved state-of-the-art results on a number of action recognition datasets including HMDB. In particular, they used the *BOF* approach for five different types of descriptors with six different types of spatio-temporal griding schemes ending up in using 480,000 dimension features. The results of 30 channels were combined in a multi-channel chi-square kernel setting. Obviously, the application of spatio-temmporal griding in feature extraction and use of many *BOF* channels can improve recognition accuracy. In our experiments, however, we focus on expressing power of late fusion and use only two dense trajectories channels, namely motion boundary histograms, and hog/hof along dense trajectories with *BOF* scheme using code books of size 4,000.

6.6.2 PPMI and Web Actions Datasets

The *Web-actions* dataset was presented in (Ikizler, Cinbis and Sclaroff, 2009). It contains a total of 2,458 images depicting 5 different human actions. We follow the same experimental setup as proposed in (Ikizler, Cinbis and Sclaroff, 2009). People playing musical instruments (*PPMI*) (Yao and Fei, 2010) is another popular action recognition dataset in still images. It introduces different challenges for recognizing the actions depicted by those images. Our experimental evaluation considers the seven classes classification task for the evaluation.

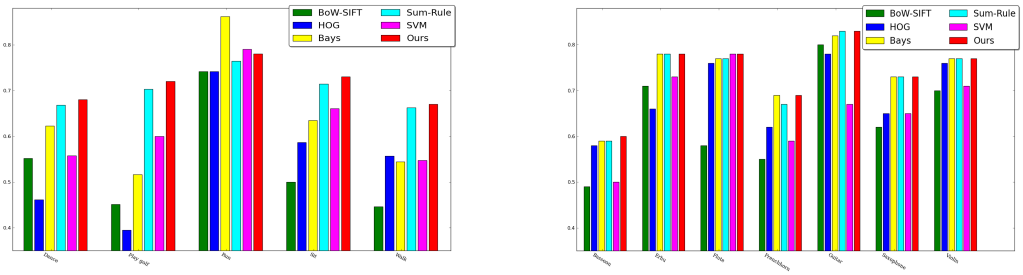
HOG and BOF Image Features

For evaluation, we harness HOG features (Dalal and Triggs, 2005) with Convolved Trilinear Interpolation (*CTI*) to distribute the effect of each pixel over its neighborhood. Our motivation for using this descriptor is that certain activities (e.g. "walk") show limited variations of pose and appearance. For the *PPMI* dataset, we crop 10% of each image's width from each side to reduce background variations. As a more flexible action representation, we use *BOF* methods with Sift local features. We gather Sift local features of different scales and construct a codebook of size 512. Local features are then encoded using Locality-constrained Linear Encoding (*LLC*) (Wang et al., 2010) and pooled into a three level spatial pyramid representation. For both features, we train our models using SVM classifiers with Gaussian kernels (Pedregosa et al., 2011).

6.7 Experimental Results

In our experiments, we use the models (feature descriptors and classifiers) described earlier and compare our late fusion approach to three different baselines: (i) the sum-rule, (ii) the Bayes method, and (iii) SVM fusion. The sum-rule is a simple fusion strategy that sums the confidence posteriors of a test sample across multiple modalities and assigns it the class label with the highest response. The Bayesian fusion, as in (Atrey, Kankanhalli and Jain, 2006), uses class-wise accuracies as probabilistic weights in the classification , i.e. $P(c|m)$ in (6.19). The SVM based fusion scheme builds a classifier on the prediction score space of different models. Note that, for all experiments, we assume that the models' confidence scores are normalized and transformed into posterior probabilities (Jain, Duin and Mao, 2000).

In order to train our classifiers, we divide the data into their standard train- and test-splits following standard guidelines or conventions for each dataset. We further divide the training data into a training set $\mathbf{D}$ and a validation set $\mathbf{V}$ to learn the model parameters along with the fusion weights w_{ij} or $P(c|m)$ discussed in Section 6.5. In case of the video dataset (HMDB) where different clips may belong to the scenes of a longer video, we use the train-test and train-validation splits which ensure that train, validation and test sets do not share clips of the same video scene. Specifically, we use the three train-test splits provided by (Kuehne et al., 2011) and divide the training data to three train-validation splits following the same guidelines.



(a) *Web-actions* (b) *PPMI7*

Figure 6.1: Class-wise accuracies of different individual and ensemble classifiers for: (a) the *Web-actions*, and (b) the *PPMI7* action image datasets

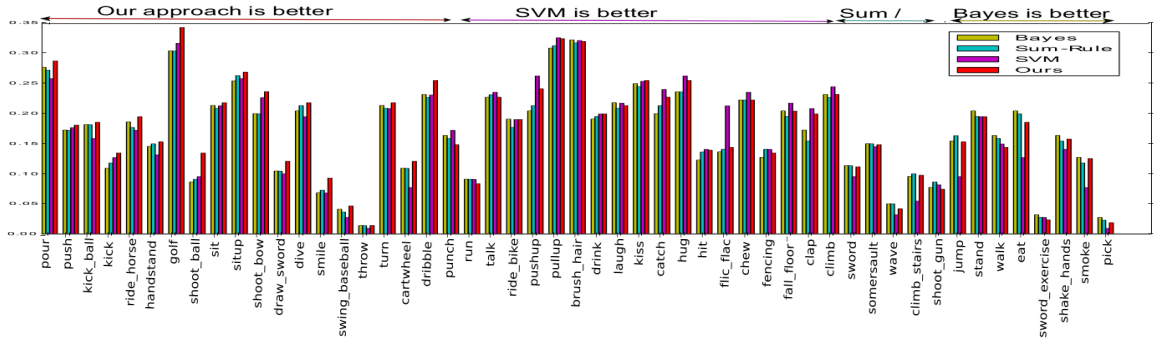


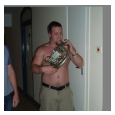
Figure 6.2: Class-wise accuracies of different individual and ensemble classifiers for the HMDB video dataset

6.7.1 Recognition Performance

Table 6.1 shows classification accuracies obtained from using our proposed fusion scheme on each dataset and compares them to three different late fusion strategies (sum-rule, Bayes and SVM-based fusion). For the HMDB dataset, features along dense trajectories show superior performance as compared to Action Bank features which show better results in comparison to STIP HOG/HOF. Recall that we are using only two channels of dense trajectories features each represented as a 4000 *BOF* vector. This is in contrast to (Wang et al., 2013b) that use 30 channels with *BOF* representations of dimensionality 480,000. Adding further channels may enhance the classification accuracy of our approach, yet its results are still superior to other methods that use moderate numbers of features. Notice further that our fusion scheme of these modalities yields the best performance of 41.83% as compared with the sum-rule (40.70%) and the SVM-rule (40.63).

Results from the *PPMI7* and the *Web-actions* image datasets show that our weighting

Table 6.2: Example images from *PPMI7* and the recognition results using individual models and different fusion methods.

							
BOF-SIFT	✓	×	✓	×	✓	×	✓
HOG	×	×	✓	✓	×	✓	×
SVM	✓	×	✓	×	×	×	×
Sum-rule	✓	×	✓	×	✓	×	✓
Bayes method	✓	×	×	✓	✓	×	✓
Ours	✓	✓	✓	✓	✓	×	✓

scheme significantly boosts classification performance (3% to 4%) as compared to SVM and Bayesian fusion. Our results also express power of the naive sum-rules towards late fusion. While SVM and Bayes rule suffer from degradation of overall performance, our method of computing class-wise probabilistic fusion weights ensures no degradation in results. Note also that our results in Table 6.1 better state-of-the-art recognition accuracies on both datasets by 6% to 8%. In particular, for the *Web-actions* dataset, we achieve 67.15% accuracy compared to the earlier best result of 61.07% (Yang, Wang and Mori, 2010). For the *PPMI7*, we report 74.0% compared to 65.7% of the Grouplets features of (Yao and Fei, 2010). Figures 6.1–6.2 plots class-wise accuracies of different models and the fusion methods. Our approach consistently outperforms other fusion schemes on all datasets where the sum-rule ranks second.

6.7.2 Distribution and Impact of Fusion Weights

As discussed above, our method for learning the stochastic weight vectors for each modality addresses the limitation of assigning fixed weights across classes for fusing the predicted scores. This can be seen, for instance, by looking at the individual class-wise classification accuracies of each modality on the *Web-actions* dataset (Figure 6.1(a)). Note that for actions that are characterized by a limited set of poses (e.g. "walk" and "sit"), employing HOG templates achieve good performance. For other actions that stand for a broad set of body poses (e.g "dance" and "play golf"), using the flexible representation of *BOF-SIFT* proves a more appropriate choice for recognition. In this sense, it is more intuitive to assign greater fusion weights to the features that best suit certain classes.

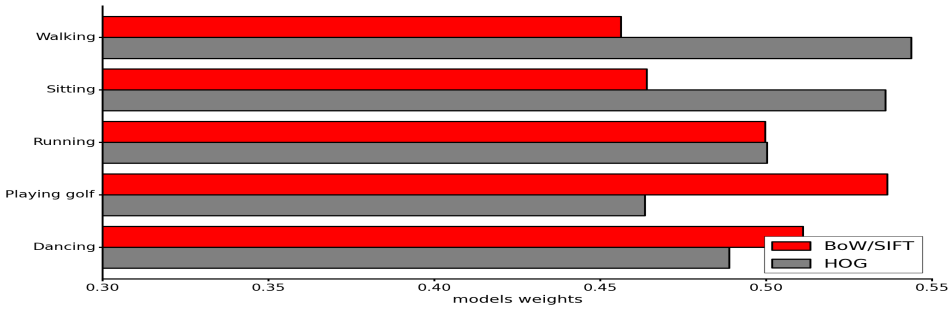


Figure 6.3: The models weights of each action for the *Web-actions* dataset.

Figure 6.3 shows the fusion weights learned for each class on the *Web-actions* dataset. These weights are stochastic vectors and therefore can be interpreted as the posterior probabilities $P(model|class)$. Notice that for the actions "walk" and "sit", the weights obtained for the HOG model are greater than those for the *BOF-SIFT* model. While for the actions "dance" and "playgolf", the *BOF-SIFT* weights dominate. However, the two models are assigned comparable weights for the action "run". Table 6.2 shows challenging images from the *PPMI7* dataset and their recognition results using *HOG*, *BOF-SIFT* and various fusion methods. In comparison to other methods, our approach is consistent and it correctly classifies the example images in most cases.

6.8 Summary

Fusing multiple representations to incorporate different sources of inter- and intra-class variations has become a paramount to human action recognition in unconstrained data. Early fusion or concatenation of multiple (sparse) feature descriptors may lead to curse of dimensionality and is computation-intensive. Therefore, the late fusion of predictions from individual classifiers is becoming a popular choice as it provides a robust integration of classifiers which (individually) perform well on different regions of instance space. In this chapter, we presented a novel and principled method for late fusion of different modalities that estimates category-wise probabilistic weights for the underlying models. Our approach is based on a quadratic objective function and employs constrained quadratic programming to determine semantically meaningful weights. Compared to existing approaches such as the sum-rule, SVM-based fusion, and Bayesian frameworks, our framework offers a flexible approach that combines favorable characteristics of these earlier methods – it considers error minimization

like in SVM-based methods and computes probabilistic weighting factors just as Bayesian approaches do. Experimental results on three challenging video and image action datasets show the prevalence and consistency of our approach. Moreover, we report 6% to 8% improvement compared to previously published results on the two image action datasets.

CHAPTER 7

Conclusions and Future Work

This dissertation has presented several approaches for human action recognition using different action representations in realistic video and image data. To conclude our work, we summarize our key achievements and discuss conclusions in Section 7.1. We then suggest possible future directions of our work in Section 7.2.

7.1 Summary of Thesis Achievements

In this section, we summarize our key achievements on the problem of human action recognition. We first present our conclusions when using the appearance representation solely for human action recognition (Chapter 3). These conclusions motivate our next research, which emphasizes the need of abstract pose-based human action representation to achieve fast and robust recognition performance (Chapter 4). In addition, we present an investigation on the benefits of different pose variants that is obtained from pose-estimation algorithms for action recognition (Chapter 5). Finally, we describe a principled approach for combining the predictions achieved from different representations. This combination is presented in a way that regards the discriminative information of the action representation for different action classes (Chapter 6).

Appearance-based human action recognition: In Chapter 3, we focused on the problem of human action recognition based solely on the actors' appearances in image data.

We have demonstrated how combining heterogeneous action features of action scenes and body pose appearances is vital to boost the recognition performance. These results were achieved through a novel multi-class classification algorithm that is based on recent advances in multiview learning using *NMF*. We also show that the resulting classification model only relies on matrix multiplications in estimating classes' posteriors. Therefore, it represents a good candidate for real-time applications where interest in classification confidence goes beyond one class to all other classes.

Discriminative pose-based approach for action recognition: Due to the heavy variation of action appearances under different views, scales, and scenes, we proposed in Chapter 4 a novel action recognition framework that is based on an abstract pose-based action representation. Action recognition frameworks can greatly benefit from pose-based representation, as it limits the effects of varying views, scenes, and scales, while focusing on the intrinsic movements of the body parts. Our discriminative framework for action recognition explicitly addresses three varying factors and solutions in pose-based action representation. These factors are:

1. Variation of humans shapes due to different actors' sizes or inaccurate measurements of the body's joint locations. To resolve this variation, we follow a part-based solution which decouples each joint feature from the body and focuses on the local characteristics of joint location, velocity, and their correlation.

2. Variation of motion of inconclusive body parts, which may differ due to different actors styles or inaccurate measurements of the body's joint locations. Therefore, we propose a discriminative part-based approach that down-weights the effects of inconclusive body parts while focusing on the conclusive ones.

3. Variation of motion of conclusive body parts for the same action, which may also differ due to different actors styles or inaccurate measurements of body's joint locations. To alleviate its effect, we softly quantize the joints features of location, velocity, and their correlation vectors to limit this variation in motion for conclusive parts.

Consequently, and unlike most previous pose-based approaches for action recognition, our framework showed efficient performance for training and testing runtimes for a range of challenging action and gesture datasets, where it achieves state-of-the-art performance in most testing scenarios.

The significance of different pose variants for action recognition: Current pose-based recovery algorithms provide different variations of pose representation including 2D, 3D, and 3D with joints' orientation. To investigate the differences among these representations for action recognition, Chapter 5 compared their performance by using the recognition framework presented in Chapter 4). In particular, our evaluation addressed the recognition gap when 2D and 3D poses are used, it also showed that 3D instead of 2D pose estimation from monocular videos has the potential to improve action recognition. Further, we demonstrated how enriching the 3D pose representation with joints' orientation benefits the recognition performance, especially for fine grain categorization of human gestures. These findings may motivate future generic pose estimation methods from monocular views to account better pose variants in their formulation.

Stochastic late fusion approach for different action modalities: Naturally, human actions are often characterized by different features of multiple data representations including action poses, scene appearances, and object shapes. Observing that these features often provide compatible and complementary information, it is natural to integrate them to achieve better performance rather than relying on a single feature representation. For this purpose, Chapter 6 proposed a late fusion approach that combines these heterogeneous representations in a way that regards the discriminative potential of each individual representation for particular action classes. Our approach presents a principled late fusion method of different modalities that estimates category-wise probabilistic weights for the underlying models. We evaluate our proposed approach on a range of challenging action recognition datasets and show that the proposed late fusion approach outperforms other late fusion techniques providing state-of-the-art classification accuracies on the benchmark data.

7.2 Future Work

Discriminative local pose-based action recognition: Our design of the discriminative framework for action recognition proposed in Chapter 4 can be enhanced by adopting the following measures:

1. **Limiting quantization artifacts:** The performance of our discriminative joints approach can be enhanced by introducing better quantization of 3D feature vectors. The current approach uses the azimuth and zenith angles for quantization. Consequently,

the size of the bins used are not uniform, especially near pole points. Alternative quantization schemes have been also proposed in (Klaser et al., 2010) where they use regular polyhedrons to provide a generic 3D quantization. This quantization scheme is bounded to the polyhedrons used, and therefore can only support up to 20 bins orientation. A recent work of (Oreifej and Liu, 2013) proposes a discriminative quantization approach of the orientations in 4D. We believe that accounting for a precise quantization approach in this regard has the potential of enhancing the overall classification of our discriminative recognition.

2. **Joint correlations features:** Encoding parts' features separately in our framework has shown efficient performance on several action recognition benchmarks. However, recent approaches (Yao et al., 2011a; Wang, Wang and Yuille, 2013; Bourdev and Malik, 2009) show that accounting for the correlation between the body parts is beneficial for action recognition. The use of such correlations can be encoded in our framework by mining informative correlations of body joints' locations, movements, appearances, and depth-imagery patterns. The mined correlations can then be invoked in a discriminative formulation that regards their role in separating among classes in the feature space for better action recognition.

3. **Further features to describe the dynamics of joints orientation:** Accounting more features for joint orientation dynamics such as angular velocity to capture further details of the joint movements. As opposed to the work in Chapter 5, we wish to explore the utility of 3D pose recovery with their joints orientation from 2D pose representation, and the expected recognition accuracy gain from such recovery. The insights may be of great interest for pose estimation research, as they provide an intuitive incentive towards 3D pose with joints orientation recovery instead of only 2D pose recovery.

Discriminative local appearance-based features for action recognition: The use of body joints as interest points for sampling part features has been frequently used in action recognition (Maji, Bourdev and Malik, 2011; Wang et al., 2012a). However, the combination of the part features used in these approaches often undermines their discriminative capacity. Following the same analogy of discriminative weighting of joints' location and motion features (Chapter 4), we suggest a similar combination of local body parts' features in a way that regards their discriminative local appearances of objects or parts' shape. Our

proposal can also be used for other action representation, including depth and silhouettes.

Temporal segmentation for better human action recognition: Many computer vision approaches for action recognition use a supervised classification approach for localizing action segments within the action sequence (Klaser et al., 2010; Zanfir, Leordeanu and Sminchisescu, 2013). However, these techniques demand training and prior knowledge of the actions, which is often not the case for challenging datasets as we have demonstrated in the *ChaLearn* dataset (Chapter 5). Moreover, the complexity of this type of localization scales linearly with the number of action classes to be found. In contrast to the supervised action temporal segmentation approach presented in Chapter 4, we reported a better performance for the *TUM* dataset when predefined splits of the action sequence were provided. These observations raise the significance of introducing temporal segmentation as a preliminary step for action recognition, or as recently proposed by Wang and Wu (2013) as a unified framework that maximizes the margin between the resulting segment features in the action space. As such, we propose two possible directions to resolve the action recognition problem for unsegmented action recognition scenarios:

1. Unsupervised segmentation of action sequences into sub-sequences which can localize possible extents of the human actions (Jones and Shao, 2014), and then perform action recognition using, for instance, our proposed action recognition framework.

2. A unified solution for action recognition and temporal segmentation using a latent SVM formulation that best localizes the action extent based on their corresponding features in a maximum margin framework (Wang and Wu, 2013).

Fusion action modalities: Fusing heterogeneous modalities to incorporate different sources of inter- and intra-class variations has become paramount to human action recognition in unconstrained data (Wang et al., 2011b; Yao, Gall and Gool, 2012; Rohrbach et al., 2012). Late fusion, in particular, became popular in this domain because it provides a robust integration of classifiers which (individually) perform well on different regions of instance space. In Chapter 6, we followed a linear weighting scheme of the modalities given for each action by a quadratic formulation that minimizes the number of misclassified samples. Despite its promising results, the provided weighting scheme is still bound to the class level. In contrast to these approaches, recent studies for late fusion have shown that accounting a sample-level (Liu et al., 2013a) or group-level weighting scheme (Liu et al., 2012) can

achieve better performance on several object detection benchmarks. Accordingly, we aim to explore varied optimization formulations than presented in Chapter 6 to investigate a sample- or group-level weighting scheme for late fusion.

Bibliography

Agarwal, A. and B. Triggs, "A local basis representation for estimating human pose from cluttered images", *ACCV*, 2006.

– "Recovering 3d human pose from monocular images", *TPAMI* 28 (2006) 44–58.

Akata, Z., C. Thurau and C. Bauckhage, "Non negative matrix factorization in multimodality data for segmentation and label prediction", *CVWW*, 2011.

Ali, S., A. Basharat and M. Shah, "Chaotic invariants for human action recognition", *ICCV*, 2007.

Atrey, P. K., M. S. Kankanhalli and R. Jain, "Information assimilation framework for event detection in multimedia surveillance systems", *Springer ACM Multimedia Systems Journal* 12.3 (2006) 239–253.

Atrey, P. K. et al., "Multimodal fusion for multimedia analysis: a survey", *Multimedia Systems* 16 (2010) 345–379.

Bach, F., G. Lanckriet and M. Jordan, "Multiple kernel learning, conic duality, and the SMO algorithm", *ICML*, 2004.

Barker, M. and W. Rayens, "Partial least squares for discrimination", *J. Chemometrics* 17 (2003).

Bauckhage, C. and T. Kaster, "Benefits of separable multilinear discriminant classification", *ICPR*, 2006.

Bauckhage, C., T. Kaster and J. K. Tsotsos, "Applying ensembles of multilinear classifiers in the frequency domain", *CVPR*, 2012.

– "Applying ensembles of multilinear classifiers in the frequency domain", *CVPR*, 2012.

– "Applying ensembles of multilinear classifiers in the frequency domain", *CVPR*, 2012.

Bay, H. et al., "SURF: speeded up robust features", *CVIU* 110.3 (2008) 346–359.

Bissacco, A. et al., "Recognition of human gaits", *CVPR*, 2001.

Blank, M. et al., "Actions as space-time shapes", *ICCV*, 2005.

Bobick, A. and J. Davis, "The recognition of human movement using temporal templates", *TPAMI* 23 (2001) 257–267.

Bourdev, L. and J. Malik, "Poselets: body part detectors trained using 3D human pose annotations", *ICCV*, 2009.

Brauer, J. and M. Arens, "Reconstructing the missing dimension: from 2d to 3d human pose estimation", *CAIP*, Spain, Málaga, 2011 25–39.

Caicedo, J. C. et al., "Multimodal representation, indexing, automated annotation and retrieval of image collections via non-negative matrix factorization", *Neurocomputing* 76 (2012) 50–60.

Campbell, L. and A. Bobick, "Recognition of human body motion using phase space constraints", *ICCV*, 1995.

Cheema, S., A. Eweiwi and C. Bauckhage, "A Stochastic Late Fusion Approach to Human Action Recognition in Unconstrained Images and Videos", *GCPR*, 2014.

– "Gait Recognition by Learning Distributed Key Poses", *ICIP*, 2012.

– "Human Activity Recognition by Separating Style and Content", *Pattern Recognition Letters* 34 (2013).

– "Who is Doing What? Simultaneous Recognition of Actions and Actors", *ICIP*, 2012.

Cheema, S. et al., "Action recognition by learning discriminative key poses", *ICCV-WORKSHOPS*, 2011.

Coates, A. and A. Ng, "The importance of encoding versus training with sparse coding and vector quantization", *ICML*, 2011.

Dalal, N. and B. Triggs, "Histograms of oriented gradients for human detection", *CVPR*, 2005.

Delaitre, V., J. Sivic and I. Laptev, "Learning person-object interactions for action recognition in still images", *NIPS*, 2011.

Deltaire, V., I. Laptev and J. Sivic, "Recognizing human actions in still images: A study of bag of features and part based representations", *BMVC*, 2010.

Ding, C., T. Li and W. Peng, "On the equivalence between non-negative matrix factorization and probabilistic latent semantic indexing", *Comput. Statatics and Data Analysis* 52 (2008) 3913–3927.

Dittrich, W., *Perception* 22 (1993) 15–20.

Dondera, R. and L. Davis, "Kernel PLS regression for robust monocular pose estimation", *CVPR-WORKSHOPS*, 2011.

Donner, R. et al., "Fast active appearance model search using canonical correlation analysis", *TPAMI* 28 (2006) 1690–1694.

Donoho, D. and V. Stodden, "When does non-negative matrix factorization give a correct decomposition into parts", *NIPS*, 2004.

Efros, A. A. et al., "Recognizing action at a distance", *CVPR*, 2003.

Escalera, X. Perez-Sala S., C. Angulo and J. Gonzalez, "A Survey on Model Based Approaches for 2D and 3D Visual Human Pose Recovery", 14 (2014) 4189 –4210.

Eweiwi, A., S. Cheema and C. Bauckhage, "Action Recognition in Still Images by Learning Spatial Interest Regions from Videos", *Pattern Recognition Letters* 51 (2015).

– "Discriminative joint non negative matrix factorization for human action classification", *GCPR*, 2013.

Eweiwi, A. et al., "Efficient Pose-based Action Recognition", *ACCV*, 2014.

Eweiwi, A. et al., "Temporal key poses for human action recognition", *ICCV-WORKSHOPS*, 2011.

Farneback, G., "Two frame motion estimation based on polynomial expansion", *SCIA*, 2003.

Felzenszwalb, P. et al., "Object detection with discriminatively trained part based models", *TPAMI* 32 (2010) 1627–1645.

Gall, J. et al., "Hough forests for object detection, tracking, and action recognition", *TPAMI* 33 (2011) 2188–2202.

Gehler, P. and S. Nowozin, "On feature combination for multiclass object classification", *CVPR*, 2009 221–228.

Gilbert, A., J. Illingworth and R. Bowden, "Action recognition using mined hierarchical compound features", *TPAMI* 33 (2011) 883–897.

– "Scale invariant action recognition using compound features mined from dense spatio temporal corners", *ECCV*, 2008.

Gorelick, L. et al., "Actions as space-time shapes", *TPAMI* 29 (2007) 2247–2253.

Gross, O. et al., "Motion interchange patterns for action recognition in unconstrained videos", *ECCV*, 2012.

Gupta, S. et al., "Nonnegative shared subspace learning and its application to social media retrieval", *KDD*, 2010.

Haj, M., J. Conzalez and L. Davis, "On partial least squares in head pose estimation: How to simultaneously deal with misalignment", *CVPR*, 2012.

hall, P., J. S. Marron and A. Neeman, "Geometric representation of high dimension, low sample size data", *J. Royal Statistical Society B* 67.3 (2005) 427–444.

Harada, T. et al., "Discriminative spatial pyramid", *CVPR*, 2011.

Harris, C. and M. Stephens, "A combined corner and edge detection", *In Alvey Vision Conference*, 1988.

He, J. et al., "Fast kernel learning for spatial pyramid matching", *CVPR*, 2008.

Hoskuldsson, A., "PLS regression methods", *Chemometrics* (1988) 211–228.

Hotelling, H., "Relations between two sets of variates", *Biometrika* 28 (1936) 321–377.

Ikizler, N., R. G. Cinbis and S. Sclaroff, "Learning actions from the web", *ICCV*, 2009.

Jain, A., R. Duin and J. Mao, "Statistical pattern recognition: a review", *TPAMI* 22 (2000) 4–37.

Jhuang, H. et al., "A biologically inspired system for action recognition", *ICCV*, 2007.

Jhuang, H. et al., "Towards understanding action recognition", *ICCV*, 2013.

Johansson, G., *Visual motion perception*, Scientific American, 1975.

Johnson, S. and M. Everingham, "Learning effective human pose estimation from inaccurate annotation", *CVPR*, 2011.

Jones, S. and L. Shao, "Linear regression motion analysis for unsupervised temporal segmentation of human actions", *WACV*, 2014 816–822.

Junejo, I. et al., "Cross view action recognition from temporal self similarities", *ECCV*, 2008.

Kaniche, M. B. and F. Bremond, "Gesture recognition by learning local motion signatures", *CVPR*, 2010 2745 –2752.

Kim, T., K. K. Wong and R. Cipolla, "Tensor canonical correlation analysis for action classification", *CVPR*, 2007.

Kittler, J. et al., "On combining classifiers", *TPAMI* 20 (1998) 226–239.

Klaser, A. et al., "Human focused action localization in video", *International Workshop on Sign, Gesture, and Activity (SGA), ECCV*, 2010.

Klingenberg, B., J. Curry and A. Dougherty, "Non negative matrix factorization: Ill posedness and a geometric algorithm", *Pattern Recognition* 42.5 (2008) 918–928.

Kovashka, A. and K. Grauman, "Learning a hierarchy of discriminative space time neighborhood features for human action recognition", *CVPR*, 2010.

Kuehne, H. et al., "HMDB: A large video database for human motion recognition", *ICCV*, 2011.

Laptev, I., "On space time interest points", *IJCV* 64 (2005) 107–123.

Laptev, I. et al., "Learning realistic human actions from movies", *CVPR*, 2008.

Lazebnik, S., C. Schmid and J. Ponce, "Beyond bags of features: spatial pyramid matching for recognizing natural scene categories", *CVPR*, 2006.

Lee, D. and H. Seung, "Learning the parts of objects by non-negative matrix factorization", *Nature* 401.6755 (1999) 788–799.

Li, M. and B. Yuan, "2D-LDA: A statistical linear discriminant analysis for image matrix", *Pattern Recognition Letters* 26 (2005) 527 –532.

Li, W., Z. Zhang and Z. Liu, "Action recognition based on a bag of 3d points", *CVPR Workshops*, 2012.

Liu, D. et al., "Sample-specific late fusion for visual category recognition", *CVPR*, 2013.

Liu, J. et al., "Learning semantic features for action recognition via diffusion maps", *CVIU* 116 (2012) 361–377.

Liu, J. et al., "Multi view clustering via joint nonnegative matrix factorization", *SDM*, 2013.

Lowe, D. G., "Distinctive image features from scale invariant keypoints", *IJCV* 60.2 (2004) 91–110.

Lucas, L. B. and T. Kanade, "An iterative image registration technique with an application to stereo vision", *Proc. Imaging Understanding Workshop*, 1981.

LXia and J.K. Aggarwal, "Spatio temporal depth cuboid similarity feature for activity recognition using depth camera", *CVPR*, 2013.

Maji, S., L. Bourdev and J. Malik, "Action recognition from a distributed representation of pose and appearance", *CVPR*, 2011.

Malinowski, M. and M. Fritz, "Learning smooth pooling regions for visual recognition", *BMVC*, 2013.

Matikainen, P., M. Hebert and R. Sukthankar, "Trajectons action recognition through the motion analysis of tracked features", *ICCV Workshop*, 2009.

Mikolajczyk, K. and C. Schmid, "An affine invariant interest point detector", *ECCV*, 2002.

Moeslunda, T., A. Hiltonb and V. Krüger, "A survey of advances in vision-based human motion capture and analysis", *CVIU* 104 (2006) 90–126.

Nandakumar, K. et al., "Likelihood ratio based biometric score fusion", *TPAMI* 30 (2008) 342–347.

Nowak, E., F. Jurie and B. Triggs, "Sampling strategies for bag of features image classification", *ECCV*, 2006.

OhnBar, E. and M. Trivedi, "Joint angles similiarities and HOG2 for action recognition", *CVPR Worshops*, 2013.

Oreifej, O. and Z. Liu, "HON4D: Histogram of oriented 4D normals for activity recognition From depth sequences", *CVPR*, 2013.

Paatero, P. and U. Tapper, "Positive matrix factorization: A non negative factor model with optimal utilization of error estimates of data values", *Environmetrics* (1994) 111–126.

Parameswaran, V. and R. Chellappa, "View invariants for human action recognition", *CVPR*, 2003.

Pedregosa, F. et al., "Scikit-learn: machine learning in python ", *Journal of Machine Learning Research* 12 (2011) 2825–2830.

Ramanan, D., "Learning to parse images of articulated bodies", *NIPS*, 2006.

Rao, C., A. Yilmaz and M. Shah, "View invariant representation and recognition of actions", *International Journal of Computer Vision* 50.2 (2002) 203–226, ISSN: 0920-5691.

Reddy, K. and M. Shah, "Recognizing 50 human action categories of web videos", *Machine Vison and Applications* 24.5 (2013) 971–981.

Ren, X., A. Berg and J. Malik, "Recovering human body configurations using pairwise constraints between parts", *ICCV*, 2005.

Rohrbach, M. et al., "A Database for fine grained activity detection of cooking activities", *CVPR*, 2012.

Rosipal, R. et al., "Kernel partial least squares regression in reproducing kernel hilbert space", *JMLR* 2 (2001) 97–123.

Sadanand, S. and J. J. Corso, "Action bank: A high-level representation of activity in video", *CVPR*, 2012.

Sala, X. et al., "A Survey on Model Based Approaches for 2D and 3D Visual Human Pose Recovery", *Sensors* 14 (2014) 4189–4210.

Salton, G. and M. McGill, *Introduction to modern information retrieval*, 1986.

Sampson, P., A. Streissguth and H. Barr F. Bookstein, "Neurobehavioral effects of prenatal alcohol: Part II. Partial least squares analysis", *Neurotoxicol Teratol* 11 (1989) 477–491.

Schmid, C., R. Mohr and C. Bauckhage, "Evaluation of interest point detectors", *IJCV* 37 (2000) 151–172.

Schölkopf, B., A. Smola and K. Müller, "Nonlinear component analysis as a kernel eigenvalue problem", *Neural Computation* 10.5 (1998) 1299–1319.

Schwartz, W. R. et al., "Human detection using partial least squares analysis", *ICCV*, 2009.

Sharma, A. and D. W. Jacobs, "Bypassing synthesis: PLS for face recognition with pose, low resolution and sketch", *CVPR*, 2011.

Sharma, G., F. Jurie and C. Schmid, "Discriminative Spatial Saliency for Image Classification", *CVPR*, 2012.

Sheikh, Y., M. Sheikh and M. Shah, "Exploring the space of a human action", *ICCV*, 2005.

Shotton, J. et al., "Efficient human pose estimation from single depth images", *TPAMI* 35.12 (2013) 2821–2840.

Shotton, J. et al., "Real time human pose recognition in parts from single depth images", *CVPR*, 2011.

Singh, S., S. A. Velastin and H. Ragheb, "MuHAVi a multicamera human action video dataset for the evaluation of action recognition methods", *AVSS*, 2010.

Sivic, J. and A. Zisserman, "Video google: A text retrieval approach to object matching in videos", *ICCV*, 2003.

Song, Y., L. Goncalves and P. Perona, "Unserpervised learning of human motion", *TPAMI* 25 (2003) 814–827.

Sun, M. and S. Savarese, "Articulated part based model for joint object detection and pose estimation", *ICCV*, 2011.

Taralova, E., F. T. Frade and M. Hebert, "Source constrained clustering", *ICCV*, 2011.

Tavakoli, A., J. Zhang and S. H. Son, "Group based event detection in under sea sensor networks", *Int. Workshop on Networked Sensing Systems*, 2005.

Taylor, C., "Reconstruction of articulated objects from point correspondences in a single uncalibrated image", *CVIU* 80 (2000) 349 –363.

Tenorth, M., J. Bandouch and M. Beetz, "The TUM kitchen dataset of everyday manipulation activities for motion tracking and action recognition", *ICCV Workshops*, 2009.

Terrades, O. R., E. Valveny and S. Tabbone, "Optimal classifier fusion in a non bayesian probabilistic framework", *TPAMI* 31 (2009) 1630–1644.

Thurau, C. and V. Hlavac, "nGrams of action primitives for recognizing human behavior", *CAIP*, 2007.

– "Pose primitive based human action recognition in videos or still images", *CVPR*, 2008.

Thurau, C. et al., "Convex non-negative matrix factorization for massive datasets", *KAIS* 29.2 (2011) 457–478.

Tran, K., I. Kakadiaris and S. Shah, "Modeling motion of body parts for action recognition", *BMVC*, 2011.

Tuytelaars, T., "Dense interest points", *CVPR*, 2010.

Vasilescu, T. Syeda-Mahmood A. and S. Sethi, "Recognizing action events from multiple viewpoints", *ICCV*, 2001.

Vemulapalli, R., F. Arrate and R. Chellappa, "Human Action Recognition by Representing 3D Skeletons as Points in a Lie Group", 2014.

Wang, C., Y. Wang and A. L. Yuille, "An approach to pose-based action recognition", *CVPR*, 2013.

Wang, H. et al., "Action recognition by dense trajectories", *CVPR*, 2011.

Wang, H. et al., "Action Recognition by Dense Trajectories", *CVPR*, 2011.

– "Dense trajectories and motion boundary descriptors for action recognition", *IJCV* 103.1 (2013) 60–79.

Wang, H. et al., "Dense trajectories and motion boundary descriptors for action recognition", *IJCV* 103 (2013) 60–79.

Wang, H. et al., "Evaluation of local spatio temporal features for action recognition", *BMVC*, 2009.

Wang, J. and Y. Wu, "Learning maximum margin temporal warping for action recognition", *ICCV*, 2013.

Wang, J. et al., "Locality-constrained linear coding for image classification", *CVPR*, 2010.

Wang, J. et al., "Mining actionlet ensemble for action recognition with depth cameras", *CVPR*, 2012.

Wang, J. et al., "Mining actionlet ensemble for action recognition with depth cameras", *CVPR*, 2012.

Wang, J. et al., "Robust 3D action recognition with random occupancy patterns", *ECCV*, 2012.

Wang, X., T. X. Han and S. Yan, "An HOG-LBP human detector with partial occlusion handling", *ICCV*, 2009.

Wanqing, L., Z. Zhengyou and L. Zicheng, "Action recognition based on a bag of 3D points", *CVPR WORKSHOPS*, 2010.

– "Action recognition based on a bag of 3d points", *CVPR WORKSHOPS*, 2010.

Wegelin, J., "A Survey of Partial Least Squares (PLS) Methods, with Emphasis on the Two-Block Case", 2000.

Weinland, D., R. Ronfard and E. Boyer, "A survey of vision-based methods for action representation, segmentation and recognition", *CVIU* 115 (2011) 224–241.

– "Free viewpoint action recognition using motion history volumes", *CVIU* 104 (2006) 249–257.

Willems, G., T. Tuytelaars and L. V. Gool, "An efficient dense and scale invariant spatio temporal interest point detector", *ECCV*, 2008.

Willems, G. et al., "Exemplar based action recognition in video", *BMVC*, 2009.

Wold, H., "Estimation of principal components and related models by iterative least squares" (1966).

– "Path models with latent variables: The NIPALS approach", *Quantitative sociology: International perspectives on mathematical and statistical model building* (1975) 307–357.

Wu, S., O. Oreifej and M. Shah, "Action recognition in videos acquired by a moving camera using motion decomposition of lagrangian particle trajectories", *ICCV*, 2011.

Xia, L. and J. K. Aggarwal, "Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera", *CVPR*, 2013.

Xu, L., A. Krzyzak and C. Y. Suen, "Methods of combining multiple classifiers and their applications to handwriting recognition", *Systems, Man and Cybernetics, IEEE Transactions on* 22 (1992) 418–435.

Yacoob, Y. and M. J. Black, "Parameterized modeling and recognition of activities", *CVPR*, 1998.

Yand, Y. and D. Ramanan, "Articulate pose estimation using flexible mixtures of parts", *CVPR*, 2011.

Yang, W., Y. Wang and G. Mori, "Recognizing human actions from still images with latent poses", *CVPR*, 2010.

Yang, X. et al., "Recognizing actions using depth motion maps-based histograms of oriented gradients.", *ACM Multimedia*, 2012.

Yang, Y. and D. Ramanan, "Articulated human detection with flexible mixtures of parts", *TPAMI* 35.12 (2013) 2878–2890.

– "Articulated pose estimation with flexible mixtures-of-parts", *CVPR*, 2011 1385–1392.

Yao, A., J. Gall and L. Gool, "Coupled action recognition and pose estimation from multiple views", *IJCV* 100.1 (2012) 16–37.

Yao, A., J. Gall and L. V. Gool, "A hough transform based voting framework for action recognition", *CVPR*, 2010.

Yao, A. et al., "Does human action recognition benefit from pose estimation", *BMVC*, 2011.

Yao, B. and L. Fei Fei, "Grouplet: A structured image representation for recognizing human and object interactions", *CVPR*, 2010.

Yao, B. and L. Fei-Fei, "Action recognition with exemplar based 2. 5D graph matching", *ECCV*, 2012.

Yao, B. et al., "Human aciton recognition by learning bases of action attributes and parts", *ICCV*, 2011.

Ye, G., D. Liu and I. Jhuo S. Chang, "Robust late fusion with rank minimization", *CVPR*, 2012.

Ye, M. et al., "A Survey on Human Motion Analysis from Depth Data", *Sensors* (2013) 149–187.

Yilmaz, A. and M. Shah, "Actions sketch: a novel action representation", *ICCV*, 2005.

Zanfir, M., M. Leordeanu and C. Sminchisescu, "The moving pose: an efficient 3D kinematics descriptor for low Latency action recognition and detection", *ICCV*, 2013.