Activity recognition in depth videos

Amir Shahroudy

2016

Amir Shahroudy. (2016). Activity recognition in depth videos. Doctoral thesis, Nanyang Technological University, Singapore.

https://hdl.handle.net/10356/69072

https://doi.org/10.32657/10356/69072

Activity Recognition in Depth Videos

Amir Shahroudy

School of Electrical and Electronic Engineering
Nanyang Technological University

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Doctor of Philosophy

2016

To Leila

Acknowledgements

First of all, I would like to acknowledge Professor Gang Wang and Dr. Tian-Tsong Ng, my supervisors, who supported me during this wonderful journey. This research was not possible without their direction, advice, encouragement, and constructive criticism.

Second, I appreciate Agency for Science, Technology and Research (A*STAR) Singapore who provided full financial support to my Ph.D. career.

Third, the director and the administration of Rapid-Rich Object Search (ROSE) Lab, Professor Alex Kot, Dr. Dennis Sng, Mr. Joseph Lim, and Ms. Qian Wang, who supported me by providing excellent research facilities and environment.

I would also like to appreciate my colleagues and group-mates who strengthened and motivated my research through fruitful discussions.

Lastly, I highly appreciate the invaluable feedback and suggestions of the anonymous examiners on my thesis.

Abstract

Introduction of depth sensors made a big impact on research in visual recognition. By providing 3D information, these cameras help us to have a view-invariant and robust representation of the observed scenes and human bodies. Detection and 3D localization of human body parts are done more accurately and more efficiently in depth maps in comparison with RGB counterparts.

Having the 3D structure of the body parts, the articulated and complex nature of human actions makes the task of action recognition difficult. One approach to handle this complexity is dividing it to the kinetics of body parts and analyzing the actions based on the partial descriptors. As the first work in this thesis, we propose a joint sparse regression based learning method which utilizes the structured sparsity to model each action as a combination of multimodal features from a sparse set of body parts. To represent dynamics and appearance of parts, we employ a heterogeneous set of depth and skeleton based features. The proper structure of multimodal multipart features are formulated into the learning framework via the proposed hierarchical mixed norm, to regularize the structured features of each part and to apply sparsity between them, in favor of a group feature selection. Our experimental results expose the effectiveness of the proposed learning method in which it outperforms other methods in all three tested datasets while saturating one of them by achieving perfect accuracy.

In addition to depth based representation of human actions, commonly used 3D sensors also provide RGB videos. It is generally accepted that each of these two modalities has different strengths and limitations for the task of action recognition. Therefore, analysis of the RGB+D videos can help us to better study the complementary properties of these two types of modalities and achieve higher levels of performance. In the second work, we propose a new deep autoencoder-based correlation-independence factorization network to separate input multimodal signals into a hierarchy of extracted components. Further, based on the structure of the features, a structured sparsity learning machine is proposed which utilizes mixed norms to apply regularization within components and group selection between them for better classification performance. Our experimental results show the effectiveness of our cross-modality feature analysis framework by achieving state-of-the-art accuracies for action classification on four challenging benchmark datasets, for which we reduce the error rate by more than 40% in three datasets and saturating the benchmark for the other one.

Recent approaches in depth-based human activity analysis achieved outstanding performance and proved the effectiveness of 3D representation for classification of action classes. Currently available depth-based and RGB+D-based action recognition benchmarks have a number of limitations, including the lack of training samples, distinct class labels, camera views and variety of subjects. In the third work, we introduce a large-scale dataset for RGB+D human action recognition with more than 56 thousand video samples and 4 million frames, collected from 40 distinct subjects. Our dataset contains 60 different action classes including daily actions, mutual actions, and medical conditions. In addition, we propose a new recurrent neural network structure to model the long-term temporal correlation of the features of each body part, and utilize them for better action classification. Experimental results show the advantages of applying deep learning methods over state-of-the-art hand-crafted features on the suggested cross-subject and cross-view evaluation criteria for our dataset. The introduction of this large scale dataset will enable the community to apply, develop and adapt various data-hungry learning techniques for the task of depth-based and RGB+D human activity analysis.

Table of contents

List of figures

List of tables

Nomenclature

Acronyms / Abbreviations

$BFGS$	Broyden–Fletcher–Goldfarb–Shanno algorithm
CCA	Canonical Correlation Analysis
CIA	Correlation-Independence Analysis
CNN	Convolutional Neural Networks
DBM	Deep Boltzmann Machine
$DCCA$	Deep Canonical Correlation Analysis
DoS	Derivatives of internal State
FTP	Fourier Temporal Pyramid
GMM	Gaussian Mixture Model
HOF	Histogram of Optical Flow
HOG	Histogram of Oriented Gradients
$HOG3D$	Histogram of 3D Gradients
$HOJ3D$	Histogram of 3D Joints
$HON4D$	Histogram of 4D Oriented Normals
$HOPC$	Histograms of Oriented Principle Components
$KCCA$	Kernel Canonical Correlation Analysis
KPM	Key Pose Motif
$L-BFGS$	Limited memory Broyden–Fletcher–Goldfarb–Shanno algorithm

LLC	Locality-Constrained Linear Coding
LOP	Local Occupancy Pattern
$LSTM$	Long Short-Term Memory
MHI	Motion History Images
$MMMP$	Multimodal Multipart learning
MP	Multipart learning
$P-LSTM$	Part-Aware Long Short-Term Memory
RNN	Recurrent Neural Network
ROP	Random Occupancy Pattern
$SSLM$	Structure Sparsity Learning Machine
$STIP$	Space-Time Interest Points
TPM	Temporal Pyramid Matching

Chapter 1

Introduction

1.1 Background

Human activity analysis is among the most challenging problems in machine vision which attracted lots of attention in recent decades. The applications of this field of research can vary from health, patient, and elderly care to large scale surveillance monitoring in public places.

Generally, the inputs in these tasks are video signals which contain millions of frame pixel values per second. Processing this tremendous amount of input is the first challenging problem for this task. Second is the loss of 3D information of the observed actions. Human actions are done in 3D space and human visual system observes and recognizes them in the same space. But traditional cameras project the visuals to a 2D frame space, hence miss some important parts of the information.

One recent and interesting solution to the above-mentioned problems are provided by 3D sensing. Recent development of depth sensors enabled us to obtain effective 3D structures of the scenes and objects. This empowers the vision solutions to move one important step towards 3D vision, *e.g.* 3D object recognition, 3D scene understanding, and 3D action recognition [2].

Having three-dimensional vision, the analysis of human actions can be further improved by tracking major joints of the bodies. A data-driven approach was proposed by Shotton *et al.* [85] to recognize human body poses in parts from depth frames in real-time. This work had a great impact and cleared the road for research in 3D human action analysis.

Microsoft Kinect 360 [29] was a cheap and easy to use depth sensor, initially provided with Microsoft Xbox 360 console for gaming purpose (Fig. 1.1) in 2011. It utilizes the real-time pose recognition algorithm of [85] and provides depth-maps

Fig. 1.1 Microsoft Kinect 360 which was released with Microsoft Xbox 360 gaming console in 2011.

and human body pose configurations, as well as RGB frames of the observed scenes. Fig. 1.2 shows an example of corresponding frames for these three modalities. Fig. 1.3 illustrates the body joints that Kinect locates and tracks in 3D space.

On 2013 an improved version of this sensor was released under the name of Microsoft Kinect v.2 (Fig. 1.4) which accompanied Microsoft's new gaming console Xbox One.

Using Kinect sensors could help us to mitigate the two above-mentioned challenges in human action analysis. Provided 3D skeletal data are very succinct and easier to use as inputs to analysis algorithms. Capturing 3D data as depth-maps instead of a 2D projection also helps to improve the analysis towards more view-invariant approaches.

1.2 Motivations

Current depth-based human activity analysis features can be divided into two major groups: joint-based (so called skeleton-based) features, and direct depth-map based ones.

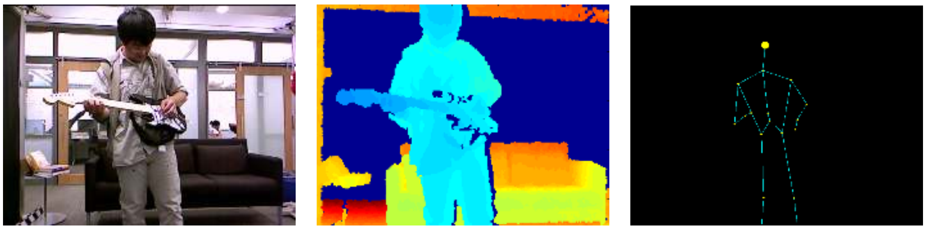


Fig. 1.2 The corresponding frames of RGB (left), depth-map (middle), and skeletal data (right) captured by Kinect 360. This example is taken from MSR-Daily Activity 3D dataset [109].

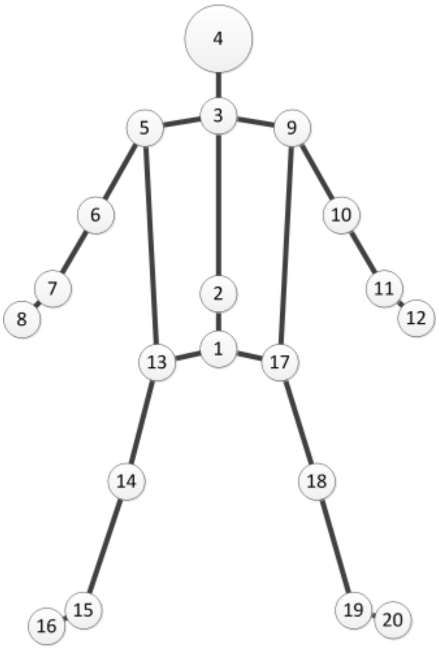


Fig. 1.3 The schema of human skeleton joints provided by Kinect.

Fig. 1.4 Microsoft Kinect v.2 which was released with Microsoft Xbox one gaming console in 2013.

Join-based methods, only use the 3D locations of body joints as inputs to the analysis algorithms. On the other hand, depth-based methods use depth videos as inputs and extract features from them for analyzing the underlying human actions.

Regarding the strengths and weaknesses of these two classes of features, we infer they are complementary to each other and to achieve higher levels of performance, we have to combine them.

The main intuition behind the work of [110] was the fact that features of few informative joints are good enough for recognizing each class of the actions. They defined “actionlet” as the combination of features of a limited numbers of joints and based on the discriminative power of each joint and each actionlet, they performed a data mining procedure to find the best actionlets for each class of the actions. They used mined actionlets as kernels in a multi-kernel multiclass SVM. Fig. 1.5 illustrates the actionlet ensemble framework.

This work motivated us to further extend their framework by applying group sparsity in a joint feature selection framework. To do so, we group the features of each part (joint) and applied L^1 norm between these groups to achieve a sparse set of active parts to represent each action class.

Mixed norms are powerful tools to inject simultaneous sparsity and coupling effects between the learning coefficients. They have been studied in a variety of fields. To achieve the sparsity between parts, we generalize this to an $L^1/L^2/L^4$ hierarchical norm.

If multiple learning tasks at hand share some inherent constituents or structures, “Multitask Learning” [14, 138, 116, 1] techniques could be globally beneficial. We also utilize the shared latent factors between different binary action classifiers of a multiclass recognition system. We apply L^2 regularization over the weights corresponding to each feature across all the tasks, followed by an L^1 between all the features at hand.

In realistic scenarios of human action recognition, adding RGB signals to the existing depth sequences is reasonable and economically efficient. The most popular depth sensors, provide RGB videos at no extra cost. The complementary properties of RGB and depth modalities for video analysis, encouraged us to move towards multimodal RGB+D activity analysis by applying deep learning. To do so, we perform correlation analysis between RGB-based and depth-based features in a every single layer of a deep correlation analysis network. We also incorporate independent components in each layer to maintain all the discriminative modality specific information, at each layer.

To evaluate depth-based and RGB+D-based activity analysis methods, there are a number of different benchmark datasets. Although each of the current datasets provide

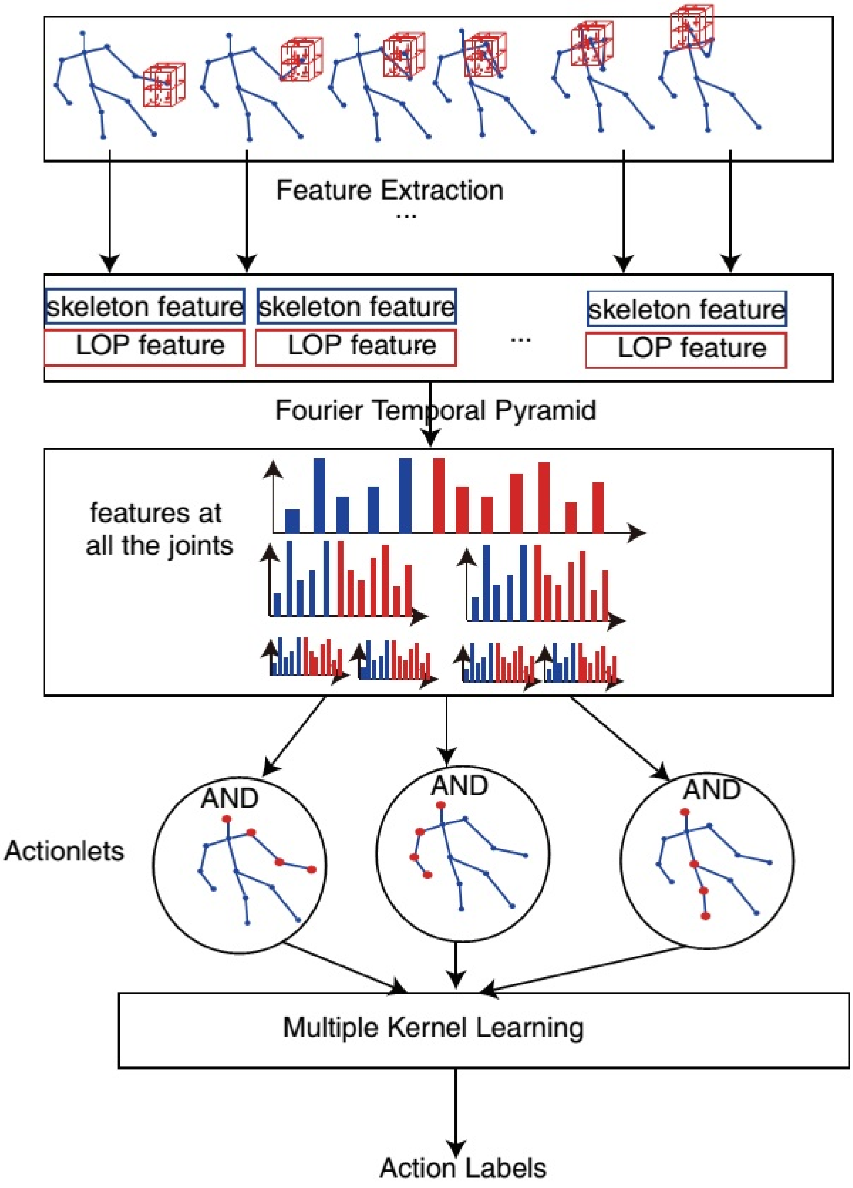


Fig. 1.5 Framework of actionlet ensemble method [110] ©2013 IEEE.

different important aspects for human activity analysis, they all have limitations on the number of samples, classes, subjects, and camera viewpoints.

The mentioned shortcomings of current datasets, prevent us from applying data-driven learning methods to the problem of RGB+D activity analysis. To mitigate these limitations, we designed a data collection protocol and collected a large-scale dataset for activity analysis in RGB+D videos. Table 5.1 shows the list of current datasets in comparison with our large-scale RGB+D action recognition dataset.

The presence of this large-scale dataset motivates us to propose a data-driven deep learning method for action analysis. We introduce a novel Recurrent Neural Network (RNN) based framework and utilized our new dataset to train and evaluate it.

1.3 Thesis Contributions

The major contributions of this thesis are:

a) We integrate the part selection process into action classification's learning phase in order to select discriminative body parts for different action classes latently, and utilize them to learn classifiers. In addition, a hierarchical mixed norm is proposed to apply the desired simultaneous sparsity and regularization over different levels of learning weights corresponding to our special multimodal-multipart features in a joint group sparsity regression framework (chapter 3).

b) Towards human activity analysis in multimodal signals, we introduce a new deep learning network for hierarchical correlation-independence factorization of RGB+D features. A structured sparsity learning machine is also proposed to explore the structure of hierarchical factorized representations for effective action classification (chapter 4).

c) The introduction of our new large-scale dataset which will enable the community to explore through new data-hungry learning frameworks for the task of RGB+D human activity analysis. The advantages of our dataset over the existing ones are: 1- many more action classes, 2- many more samples for each action class, 3- much more intra-class variations (poses, environmental conditions, interacted objects, age of actors, ...), 4- more camera views, 5- more camera-to-subject distances, and 6- used Kinect v.2 which provides more accurate depth-maps and 3D joints, especially in a multi-camera setup compared to the previous version of Kinect.

In addition to the introduction of the dataset, we propose a novel action learning method based on Long Short-Term Memory (LSTM) networks and evaluate it alongside some baseline methods on our new dataset (chapter 5)..

1.4 Organization of the Thesis

This thesis is organized as follows: Chapter 1 introduces the motivations and contributions of the thesis. Chapter 2 reviews the related works in the literature. In chapters 3, our "multimodal-multipart" depth-based action learning framework is described. Chapter 4, presents our "deep correlation-independence analysis" method for multimodal RGB+D activity learning. The description of "NTU-RGB+D action dataset" and our part-based recurrent learning framework are provided in chapter 5. Finally, chapter 6 concludes the thesis and shows our future research directions.

Chapter 2

Literature Review

During recent years, a decent number of methods were proposed for human activity analysis in depth-based signals. In this chapter, we review the literature of notable features, learning algorithms, techniques, and benchmark datasets in depth-based human activity analysis.

2.1 3D Human Activity Analysis

Features extracted from depth signals for action analysis can be classified into two major classes. The first are skeleton based features, which extract information from the provided 3D locations of body joints on each frame of the sequence. Second are the methods which extract features directly from depth-maps. In this section we review these two groups of methods.

2.1.1 Skeleton-based Activity Recognition Methods

Essentially, skeletons have a very succinct and highly discriminative representation of the actions. In ideal noiseless case, they are also fully view-invariant since they represent the sequences of full body poses over time. These properties, makes skeleton-based methods to be more convenient for human action analysis.

Yang and Tian [124] utilized skeletal data to extract 3D difference vectors between all pairs of joint points as feature representation at each frame. They took the differences of these features between consecutive frames and the their changes compared to the initial frame into account. To reduce the dimension of these long feature vectors, they applied PCA to extract their “EigenJoints”. In the classification step, they shown

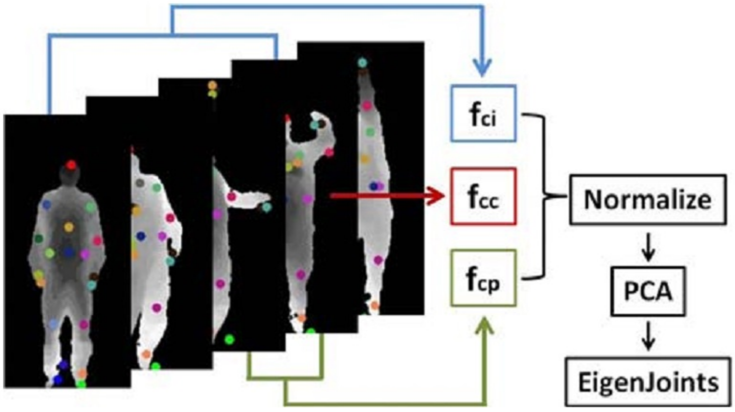


Fig. 2.1 Framework of EigenJoints method [123] 2012IEEE.

nïve-bayes-nearest-neighbor classifier could outperformed SVM. Fig. 2.1 illustrates this framework.

Ohn-Bar and Trivedi [65] combined second order joint-angle similarity representations of skeletons with a modified two step HOG feature on spatio-temporal depth maps to build global representation of each video sample and utilized a linear SVM to classify the actions.

Histogram of 3D joints (HOJ3D) was proposed by Xia *et al.* [121]. Unlike the above mentioned work, they kept one of the joints (middle hip joint) as the reference and all other joints were considered as the difference vector from the reference. To do so, they put the reference joint in the center point of spherical coordinates and divided the whole sphere into 84 sub-volumes (see Fig. 2.2). Histograms of the joint points inside these bins were smoothed using a probabilistic voting scheme and linear discriminant analysis was utilized to build a robust discriminative set of features. The quantized feature vectors were passed into HMM to model the temporal changes and final action classification.

A dictionary learning approach to encoding different skeleton variations was proposed by Luo *et al.* [60]. To have discriminative codes, they learned class specific dictionaries for each action label. They injected group sparsity and geometry constraints into their coding framework in order to have more consistent and robust features. They claimed elastic net regularizer [141] could apply group sparsity within their dictionary learning method, and geometric constraint [26] could force similar samples from the same class to have alike codes. Dictionary codes then went through a 3-level Temporal Pyramid Matching (TPM) [46] to build holistic discriminative representations of input samples. Finally they applied SVM classifier as the final step.

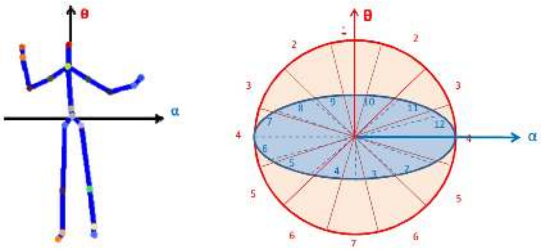


Fig. 2.2 Spherical coordinates for histogram calculation in HOJ3D method [121] ©2012 IEEE.

Vemulapalli *et al.* [100] represented different skeleton configurations as points on a Lie group. Actions as time-series of skeletal configurations, were encoded as curves on this manifold.

An interesting work was done on action recognition using both skeleton and depth map features in [110]. The difference vectors of all pairs of joints was considered as skeleton based features and local occupancy patters which where 3D histograms of depth map pixels around each joint were took as depth map representation. For each frame, all of these features were concatenated and after putting all the frames of each sample together, each of these feature elements had a 1D signal over temporal axis. To analyze the dynamics of each feature, they proposed to apply short time Fourier transformation [66] over 3 layers of a temporal pyramid. The output coefficients were called Fourier Temporal Pyramid (FTP) features which were highly robust against the natural noise of the depth maps and miscalculations of skeleton joints. This set of coefficients represented discriminative information about temporal changes of each input feature, so it could be a useful carrier of information about the movements and the action done in the input video sample. These ideas were followed by the proposal of actionlet, which was defined as a conjunctive (AND) combination over the Fourier temporal pyramid features. A mining technique was introduced to find the most discriminative actionlets for each class which could play the role of SVM kernels in a multi-class multi-kernel learning scheme. Figure 1.5 illustrates the general framework of this method.

Evangelidis *et al.* [24] split the body to groups of joint quads and represented each quad by a 6D vector. They utilized Fisher vectors to aggregate the quads as global representation of the action samples.

Meng *et al.* [61] proposed a real-time action recognition method by applying random forest classifier on a set of distance values between the body joints and interacted objects.

Wang and Wu [112] applied max-margin time warping to match the descriptors of skeletons over the temporal axis and learn phantom templates for each action class.

Hu *et al.* [34] proposed dynamic skeletons as Fourier temporal pyramids of spline-based interpolated skeleton points and their gradients, and HOG-based dynamic color and depth patterns to be used in a RGB+D joint-learning model for action classification.

Wang *et al.* [101] modeled the actions as ordered sets of poses, or so called Key Pose Motifs (KPM), to cope with the intra-class variations and small-sized training data which are the challenges of 3D action recognition. They proposed a novel mining technique to discover class-specific KPMs in the training data. To classify a query action sample, they simply compared its matching scores with the mined KPMs of all action classes.

The applications of recurrent neural networks for 3D human action recognition were explored very recently [99, 22, 139].

Du *et al.* [22] applied a hierarchical RNN to discover common 3D action patterns in a data-driven learning method. They divided the input 3D human skeletal data to five groups of joints and fed them into a separated bidirectional RNN. The output hidden representation of the first layer RNNs were concatenated to form upper-body and lower-body mid-level representation and these were fed to the next layer of bidirectional RNNs. The holistic representation for the entire body was obtained by concatenating the output hidden representations of these second layer RNNs and it was fed to the last RNN layer. The output hidden representation of the final RNN was fed to the softmax classifier for action classification.

Differential RNN [99] added a new gating mechanism to the traditional LSTM to extract the derivatives of internal state (DoS). The derived DoS was fed to the LSTM gates to learn salient dynamic patterns in 3D skeleton data.

The work of [139] introduced an internal dropout mechanism applied to LSTM gates for stronger regularization in the RNN-based 3D action learning network. To further regularize the learning, a co-occurrence inducing norm was added to the network's cost function which enforced the learning to discover the groups of co-occurring and discriminative joints for better action recognition.

The drawbacks of skeletal representation in action recognition are two-fold. The presence of noise in depth maps and occlusion of body parts bounds the reliability of this type of features. Another major deficiency of skeleton data is their incapacity to represent the interactions of the body with other objects which is crucial for activity interpretation.

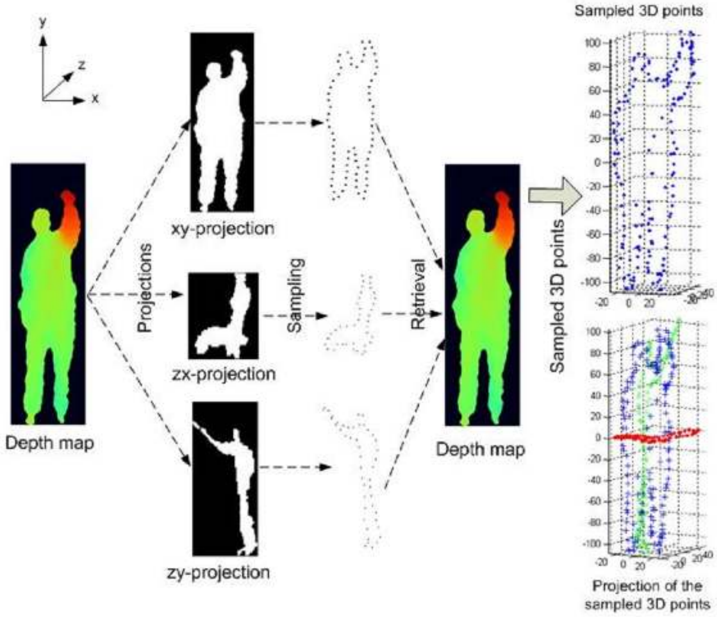


Fig. 2.3 3D point sampling method for salient posture representation ©2010 IEEE [51]

2.1.2 Depth-based Activity Recognition Methods

The other group, consists of features which are extracted directly from depth-maps. Depth-maps consist of matrices of depth measurements from the observed scenes by the acquisition sensor. Putting frames together over time axis, we have a sequence of depth maps which carry 3D information about the observed scenes. To analyze human actions in depth sequences, we first need to extract informative and discriminative features, similar to RGB counterparts and use them as inputs to a feature analysis and classification algorithm to decide about the observed actions.

One of the early attempts on depth-based analysis was proposed by Li *et al.* [51]. They built an action graph in which each node was a salient posture and actions were represented as paths through graph nodes. They proposed a novel 3D point sampling technique to build the salient postures and utilized Gaussian Mixture Models (GMM) to model the statistics of the points on them. As illustrated in Figure 2.3, the depth map was projected onto x-y, y-z and z-x planes and sampling was done over projected points at equal distance along the contours of the projections.

To alleviate the occlusion and noise issues which are natural challenges in depth-map analysis, Wang *et al.* [108] proposed Random Occupancy Pattern (ROP) features, extracted from 4D randomly sampled sub-volumes of the sequence. A simple weighted sampling method was also suggested to give more chance to discriminative sub-volumes to be included. Finally they applied an elastic-net regularization [141] to find

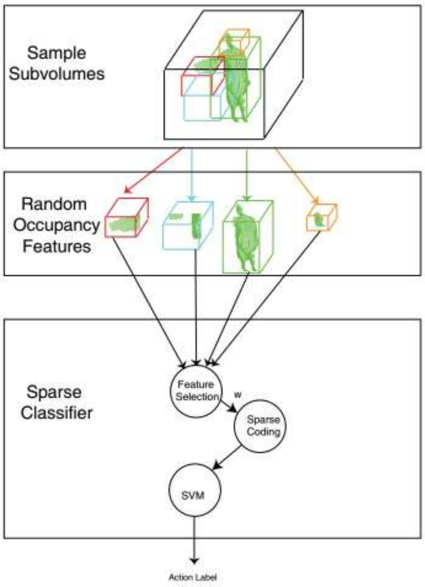


Fig. 2.4 The framework of Random Occupancy Features method proposed in [108] ©2012 Springer.

the most discriminative subset of features for action recognition. Figure 2.4 shows the entire framework of this method.

Histograms of 3D point clouds in a 3D grid around each of the joints, or so called Local Occupancy Patterns (LOP) was proposed by Wang *et al.* [110], to be concatenated to skeleton-based features for action classification using the "actionlet ensemble" framework.

Depth-map surface normals were shown to be very powerful representations of body movements over depth signals by Oreifej *et al.* [67]. They calculated the 4D oriented normals (X-Y-depth-time) from depth maps and accumulated them on spatio-temporal cubes as quantized histograms over 120 vertices of a regular polychoron. This idea was a generalization of Histogram of 3D Gradients (HOG3D) [37] to four dimensional depth videos. They quantized the 4D normal vectors of depth surfaces by taking their histograms (Histogram of 4D Oriented Normals or HON4D) over the vertices of a 4D regular polychoron, which were shown to be highly informative for action classification. Fig. 2.5 depicts the extraction framework of HON4D.

Wang *et al.* [112] added local HON4D [67] into joint features to learn a max-margin temporal warping based action classifier.

Space-Time Interest Point (STIP) detection described by Histogram of Oriented Gradients (HOG) [20] and Histogram of Optical Flow (HOF) was originally proposed for recognition purposes on RGB videos [45], but Ni *et al.* [63] showed this could be

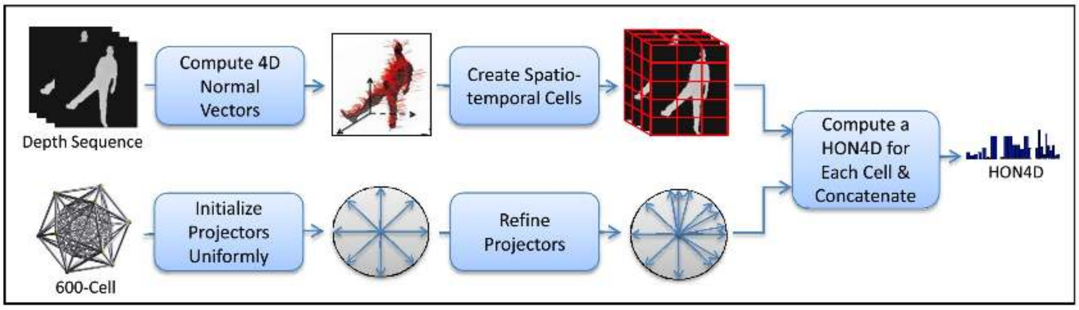

Depth Sequence
Compute 4D Normal Vectors
Create Spatio-temporal Cells
Compute a HON4D for Each Cell & Concatenate
HON4D
600-Cell
Initialize Projectors Uniformly
Refine Projectors

Fig. 2.5 Flowchart of HON4D extraction from depth maps [67] ©2013 IEEE.

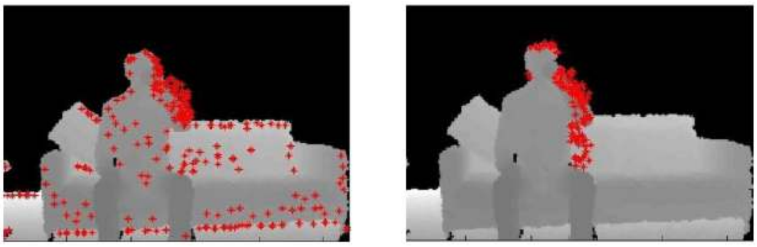


Fig. 2.6 The effect of noise supression on spatio-tempral interest point detection [120]. Left shows the results without correction function and right shows the corrected version. ©2013IEEE

easily generalized into RGB+D signals. To improve the discrimination of descriptors, they generalized the idea of Motion History Images (MHI) [8] over depth maps.

Since Kinect depth signals are noisy, STIP detection needs a noise-suppression preprocessing step in depth maps. Xia and Aggarwal [120], proposed a depth signal noise-suppression method to fill the zero valued holes in the sequence. Fig. 2.6 shows the effect of their proposed denoising method on detections of STIP. To encode the interest points, a similarity feature for depth cuboids was also introduced. They learned a dictionary of features and applied bag of words scheme to transform samples into histograms of codewords. To further improve their results, they utilized a discriminative feature mining technique.

Rahmani *et al.* [73] achieved higher levels of robustness against viewpoint variations by using Histograms of Oriented Principle Components (HOPC) of 3D point clouds. In [70] local histograms of oriented gradients from spatio-temporal cells of action videos were aggregated by Locality-Constrained Linear Coding (LLC). Yang and Tian [125] proposed supernormal vectors as aggregated dictionary-based codewords of four-dimensional normals over space-time grids. Lu *et al.* [58] applied τ test based binary range-sample features on depth maps and achieved robust representation against noise, scaling, camera views, and background clutter. Random decision forests were utilized for learning and feature pruning over a combination of depth and skeleton-based features by [72].

Convolutional Neural Networks (CNN) have also been utilized for depth based action recognition by Wang *et al.* [117]. The fusion of various depth based features is also studied by Zhu *et al.* [140]. Rahmani and Mian [74] introduced a nonlinear knowledge transfer model to transform different views of human actions to a canonical view. To apply ConvNet-based learning to this domain, [75] used synthetically generated data and fitted them to real mocap data. Their learning method was able to recognize actions from novel poses and viewpoints.

2.2 RGB+D Fusion for Action Recognition

Although depth-only methods achieved outstanding performances in human activity analysis evaluations, in real scenarios we usually have RGB videos in addition to depth. Because of the complementary properties of these two signals, multimodality analysis of RGB+D action videos also attracted some research attention recently.

Ni *et al.* [63] introduced a RGB+D fusion method by concatenating depth descriptors to RGB based representations of STIP points. A structured sparsity based fusion of local RGB+D descriptors is also proposed by [84]. Liu and Shao [56] introduced a genetic programming framework to improve the RGB and depth descriptors and their fusion simultaneously through an iterative evolution. Song *et al.* [88] solved the problem of RGB+D action recognition by utilizing RGB information for better tracking of interest point trajectories and describe them by depth-based local surface patches. Hu *et al.* [34] proposed a heterogeneous multitask feature learning framework to mine shared and modality-specific RGB+D features. Kong and Fu [38] applied projection matrices to the common and independent spaces between RGB and depth modalities. They learned their model by minimizing the rank of their proposed low-rank bilinear classifier.

Upon successful analysis of human actions in a single camera, human tracking and person re-identification techniques [95, 96, 98, 97, 57] can be utilized to extend the analysis over a network of cameras.

2.3 Depth-based Activity Analysis Benchmark Datasets

After the release of Microsoft Kinect [134], several datasets are collected by different groups to perform research on 3D action recognition and to evaluate different methods in this field. In this section we briefly review some of the publicly available 3D activity analysis benchmark datasets and the standard evaluation protocols used on them. Here we introduce a limited number of the most famous ones. For a more extensive list of current 3D activity analysis datasets and methods, readers are referred to these survey papers [132, 2, 17, 28, 59, 127, 12].

2.3.1 Datasets

MSR-Action3D dataset

MSR-Action3D dataset [51] was one of the earliest ones which opened up the research in depth-based action analysis. The samples of this dataset were limited to depth sequences of gaming actions. It provided depth sequences and skeleton information of 567 samples for 20 gaming-based actions classes: *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,* and *pick up & throw*. Later the body joint data was added to the dataset. Actions are performed by 10 different subjects, two or three times each. Evaluations are applied over a fixed cross-subject setting; Odd numbered subjects are taken for training and even numbered ones for testing. Depth sequences in this dataset have clean background which eases the recognition task. Later the body joint data was added to the dataset. Joint information includes the 3D locations of 20 different body joints in each frame. A decent number of methods are evaluated on this benchmark and recent ones reported close to saturation accuracies [60, 58, 83]. Fig. 2.7 shows the depth-maps from a sample in MSR-Action3D dataset.

CAD-60 and CAD-120 datasets

CAD-60 [90] and CAD-120 [39] contain RGB, depth, and skeleton data of human actions. The special characteristic of these datasets is the variety of camera views. Unlike most of the other datasets, camera is not bound to front-view or side-views. However, the limited number of video samples (60 and 120) is their downside. Fig. 2.8 depicts the 3 modalities of the data from different samples in these datasets.

RGBD-HuDaAct dataset

RGBD-HuDaAct [63] is a large size benchmarks for human daily action recognition in RGB+D. This dataset includes 1189 RGB+D video sequences from 13 action classes:*exit the room, make a phone call, get up from bed, go to bed, sit down, mop*

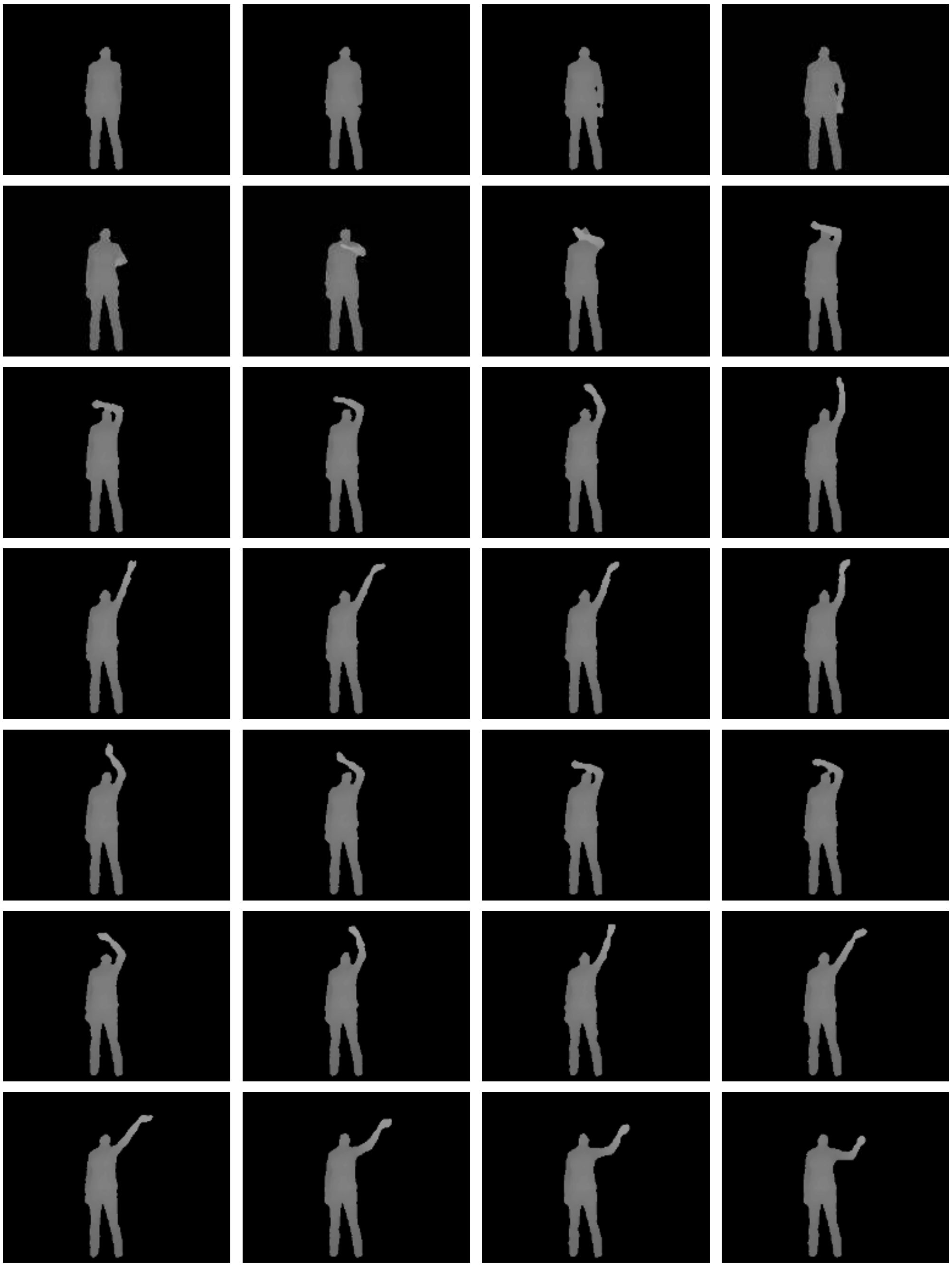


Fig. 2.7 Illustration of the depth-maps from a sample in MSR-Action3D dataset. This sample shows the action “high arm wave”.

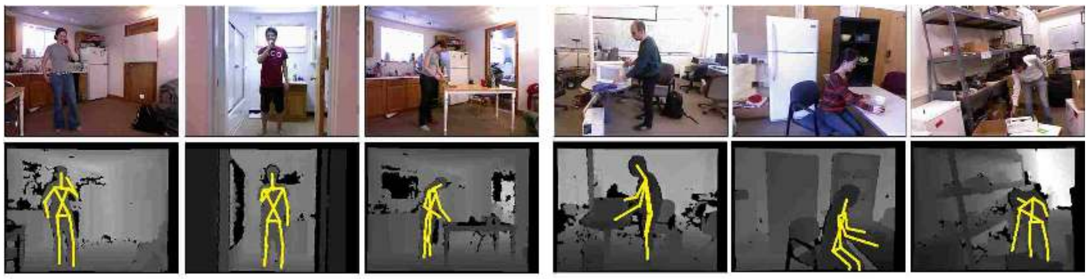


Fig. 2.8 Illustration of the RGB frames and corresponding depth-maps and skeletons from different samples in CAD-120 dataset.

floor, stand up, eat meal, put on jacket, drink water, enter room, take off jacket, and *background activity*. Actions are performed by about 30 human subjects. This dataset provided high variation in time lengths. The standard evaluation on this dataset is defined on a leave-one-subject-out cross-validation setting. The special characteristic of this dataset was that they synced and aligned the RGB and depth channels which enabled local multimodal analysis of RBGD signals[1]. Fig. 2.9 shows the depth-maps and corresponding RGB frames from a sample in RGBD-HuDaAct dataset. Fig. 2.10 depicts the RGB sample frames of different action classes in this dataset..

MSR-Daily Activity 3D dataset

According to its intra-class variations and choices of action classes, MSR-Daily Activity 3D dataset [109], is one of the most challenging benchmarks for action recognition in depth sequences. It contains RGB, depth, and skeleton information of 320 action samples, from 16 classes of daily activities in a living room: *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,* and *sit down*. Each activity is done by 10 distinct subjects in two different ways and evaluations are applied over a fixed cross-subject setting; first five subjects are taken for training and others for testing. Unlike other datasets, MSR-DailyActivity has a more realistic variation within each class. Subjects used both hands randomly to do the activities, and samples of each class are captured in different poses. Small number of samples and the fixed viewpoint of the camera are the limitations of this dataset. Recently reported results on this dataset also achieved very high accuracies [58, 34, 60, 82]. Fig. 2.11 shows the depth-maps and corresponding RGB frames from a sample in

[1] We emphasize the difference between RGBD and RGB+D terms. We suggest to use RGBD when the two modalities are aligned pixel-wise, and RGB+D when the resolutions of the two are different and frames are not aligned.

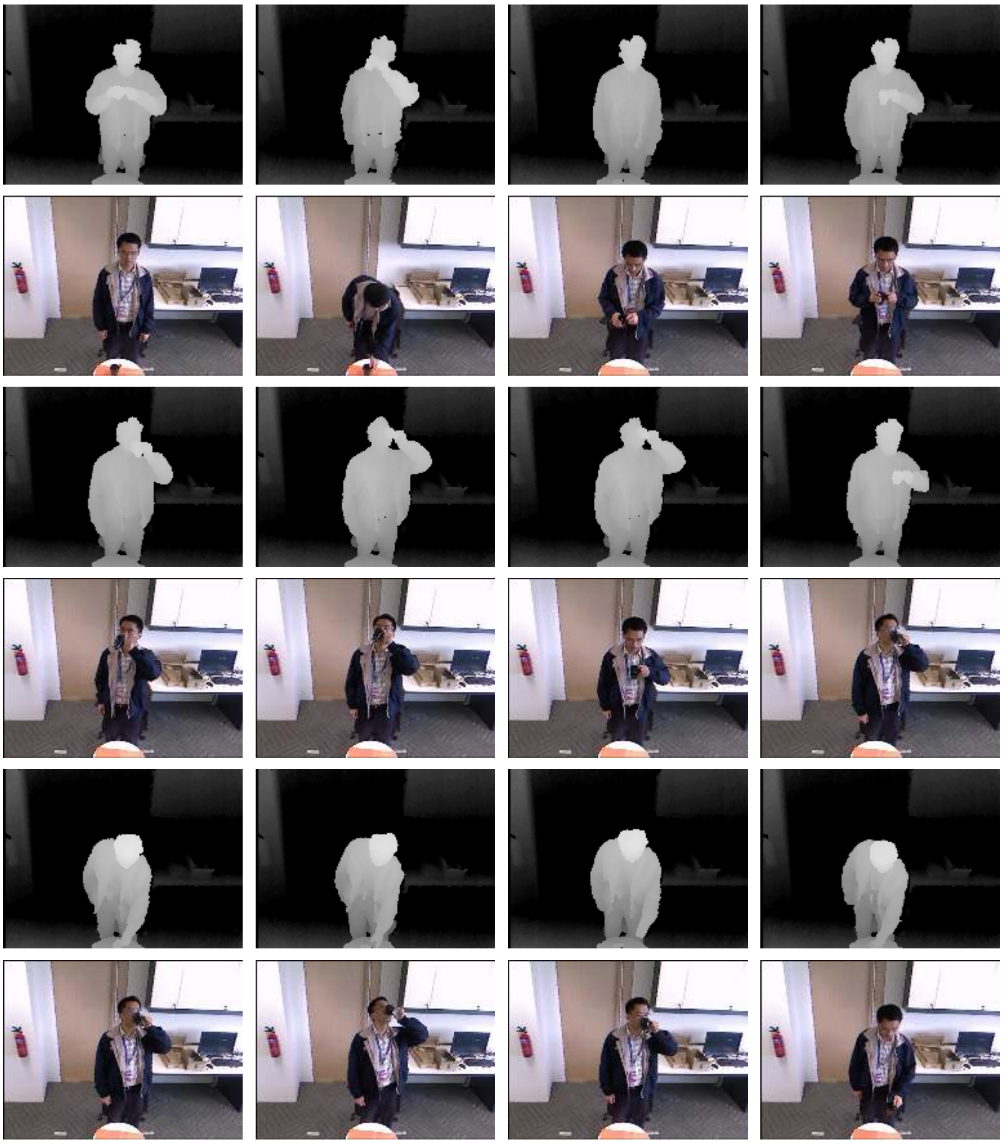


Fig. 2.9 Illustration of the depth-maps and corresponding RGB frames from a sample in RGBD-HuDaAct dataset. This sample shows the action “drink water”.

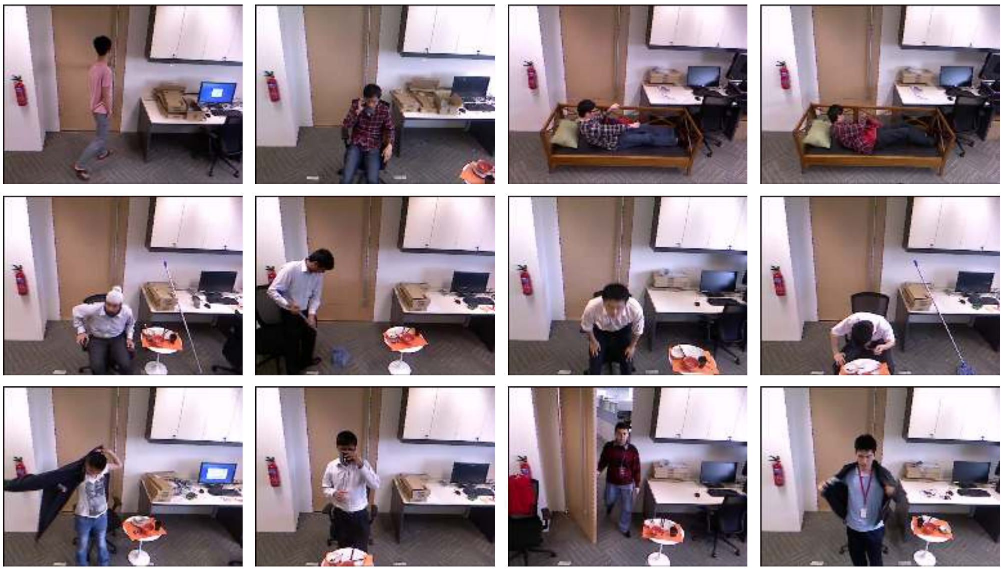


Fig. 2.10 RGB sample frames of 12 different actions in RGBD-HuDaAct dataset.

MSR-Daily Activity 3D dataset. Fig. 2.12 illustrates the RGB sample frames of 16 different actions in this dataset.

UTKinect dataset

UTKinect dataset [121] also provided synchronized RGB, depth and skeleton information. Action classes include: *walk, sit down, stand up, pick up, carry, throw, push, pull, wave hands, clap hands.*, which were peformed by 10 different subjects, twice per subject. Overall, 200 action samples are collected. They introduced more intra-class variations by asking the subjects to perform the actions differently. Very recent methods [55] achieved close to saturation results on this dataset. Fig. 2.13 depicts the two modalities of the data from different samples in these datasets.

3D Action Pairs dataset

To emphasize the importance of the temporal order of body poses on the meaning of the actions, [67] proposed 3D Action Pairs dataset. It covers 6 pairs of similar actions: *Pick up a box/Put down a box, Lift a box/Place a box, Push a chair/Pull a chair, Wear a hat/Take off a hat, Put on a backpack/Take off a backpack,* and *Stick a poster/Remove a poster*. The only difference between each pair is their temporal order so they include similar skeleton, poses, and object shapes. Each action is performed by 10 subjects, 3 times. First five subjects are taken for testing and others for training. Due to the fewer number of the action classes and absence of intra-class variations, this dataset

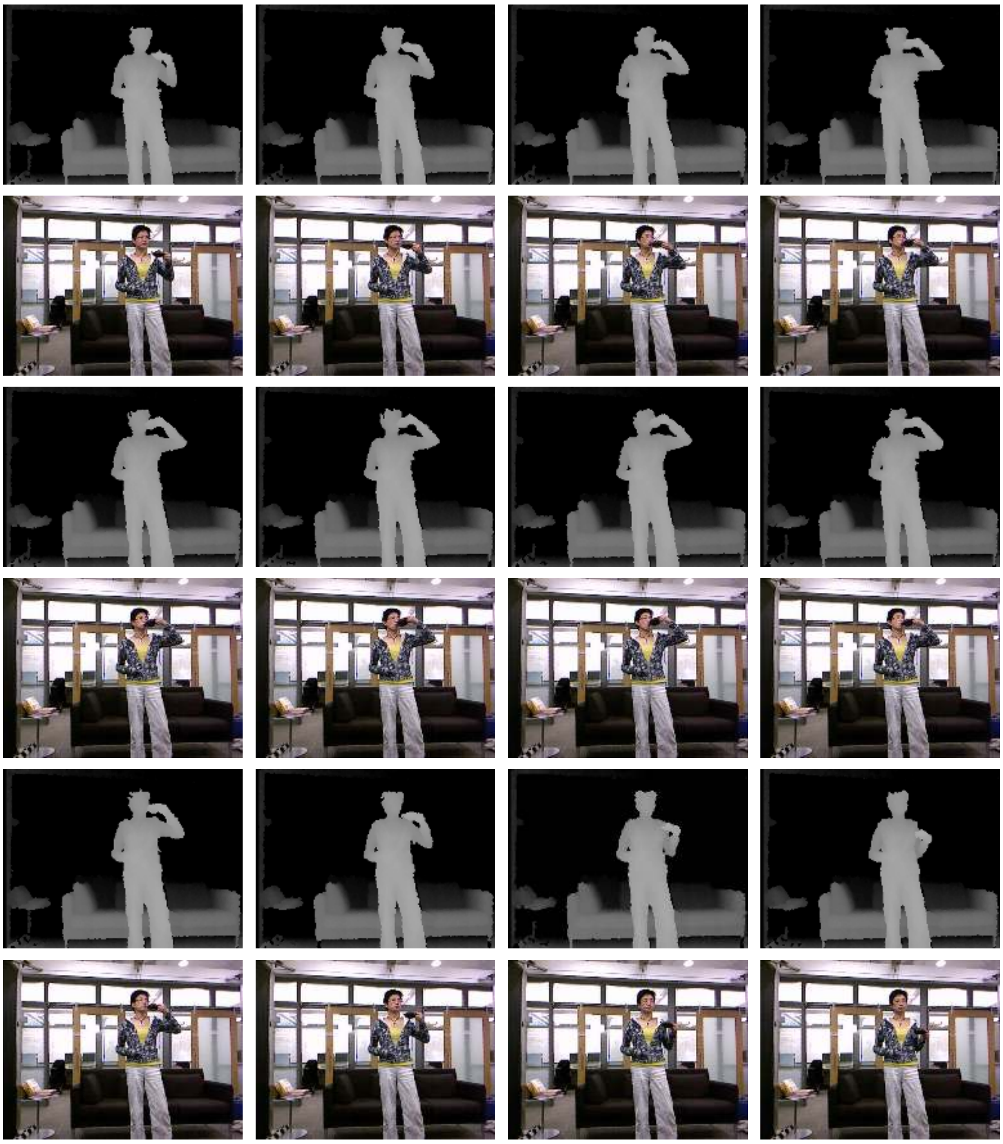


Fig. 2.11 Illustration of the depth-maps and corresponding RGB frames from a sample in MSR-Daily Activity 3D dataset. This sample shows the action “drink”.

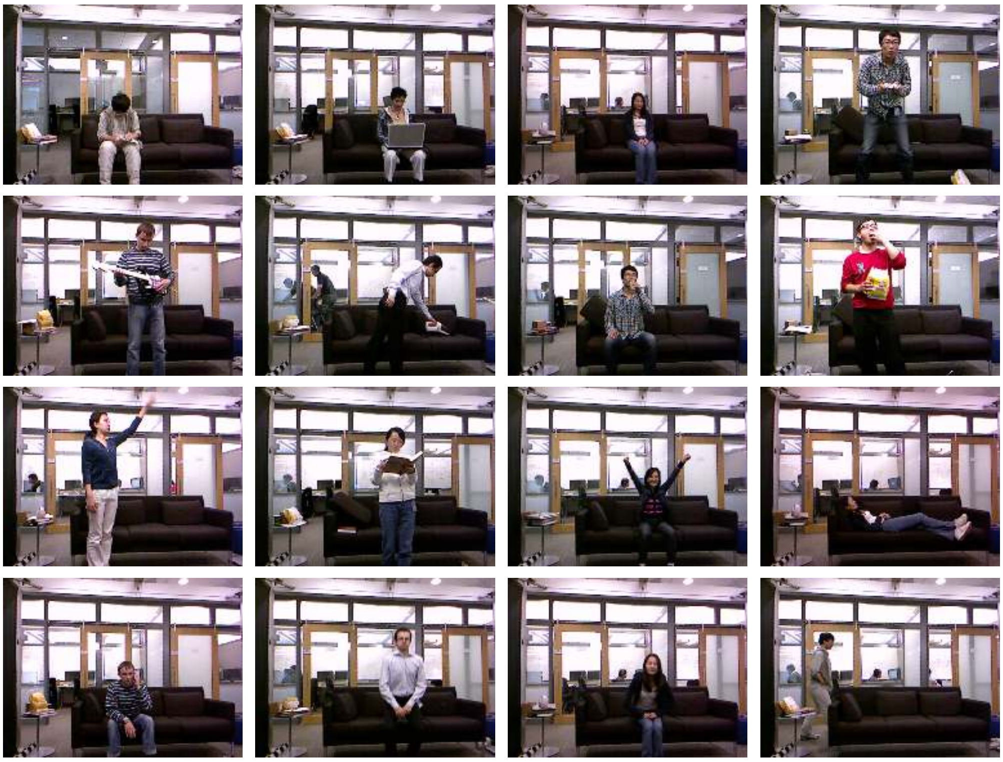


Fig. 2.12 RGB sample frames of 16 different actions in MSR-Daily Activity 3D dataset.

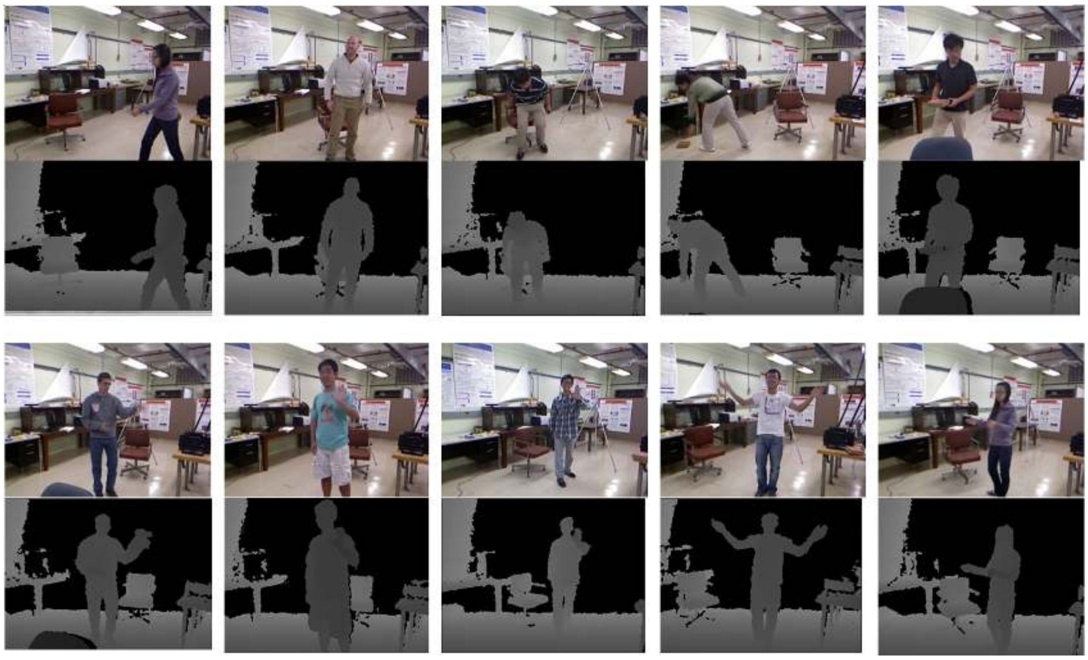


Fig. 2.13 Illustration of the RGB frames and corresponding depth-maps from different samples in UTKinect dataset.

is one of the easiest benchmark among depth based action recognition datasets and other methods already achieved very high accuracies on it. State-of-the-art methods [38, 83, 82] achieved close to saturation accuracies on this benchmark. Fig. 2.14 shows the depth-maps and corresponding RGB frames from a sample in 3D Action Pairs dataset. Fig. 2.15 depicts RGB sample frames of 12 different actions in this dataset.

Online RGB+D Action dataset

Online RGB+D Action dataset [128] is another RGB+D benchmark for action recognition. Unlike most of the other RGB+D benchmarks, this dataset is collected in different locations and provides a cross-environment evaluation setting. It includes samples of 7 daily action classes: *drinking, eating, using laptop, reading cellphone, making phone call, reading book*, and *using remote*. For the recognition task, it provides videos of 24 actors. Each actor performs each of the actions twice. Overall, this dataset include 336 RGB+D video samples. Fig. 2.16 shows the depth-maps and corresponding RGB frames from a sample in Online RGB+D Action dataset. Fig. 2.17 illustrates RGB sample frames of 7 different actions in the two different environments from this dataset

ACT4^2 dataset

ACT4^2 [18] dataset was captured using 4 Kinect cameras at the same time. They provided samples of 14 action classes: *collapse, drink, make phone call, mop floor, pick up, put on, read book, sit down, sit up, stumble, take off, throw away, twist open, wipe clean*, performed by 24 subjects. This dataset captured the outstanding 6844 number of action samples, which was among the biggest numbers of samples between the depth-based human actions dataset.

Other datasets

Multiview 3D event [118] and Northwestern-UCLA [111] (Fig. 2.18) datasets also used more than one Kinect cameras at the same time to collect multi-view representations of the same action, and scale up the number of samples.

SBU Kinect Interaction [131], LIRIS Human Activities [119], G3Di [7], and Office Activity [114] datasets are depth-based benchmarks which also include activities of interactions between two or more subjects.

It is worth mentioning, there are more than 40 datasets specifically for 3D human action recognition. The survey of Zhang *et al.* [132] provided a great coverage over

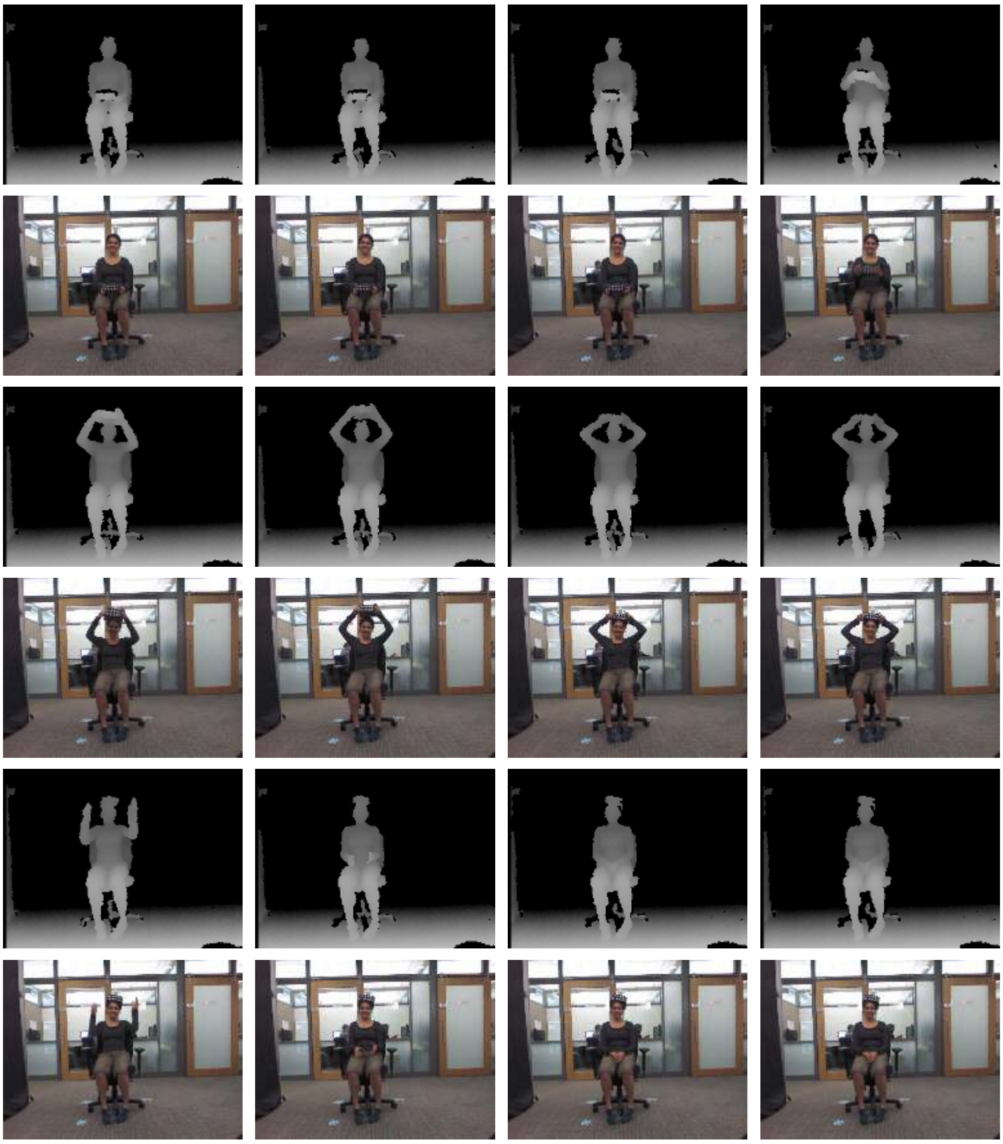


Fig. 2.14 Illustration of the depth-maps and corresponding RGB frames from a sample in 3D Action Pairs dataset. This sample shows the action “wear a hat”.

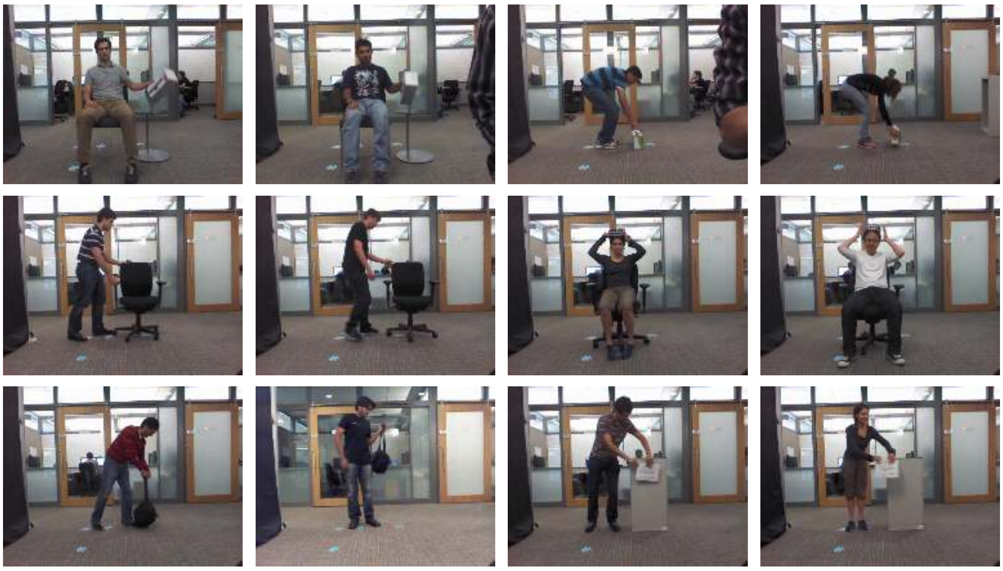


Fig. 2.15 RGB sample frames of 12 different actions in 3D Action Pairs dataset. Each pair of the actions are similar in appearance but different over temporal axis.

the current datasets and discussed their characteristics in different aspects. They also provided the best performing methods for each dataset.

Although each of the current datasets provided important challenges of human activity analysis, they have limitations in some aspects. Table 2.1 shows the comparison between some of the current datasets with our large-scale RGB+D action recognition dataset.

2.3.2 Evaluation Protocols

Various benchmark datasets determined different protocols as their standard evaluation criteria. Below are the most commonly used evaluation protocols.

Cross-Subject Protocol

In cross-subject evaluation protocols, samples are divided to fix training and testing sets based on their subject IDs. As an example, in MSR-Daily Activity 3D dataset, samples of subjects one to five were assigned for training and the remaining samples were kept out for testing.

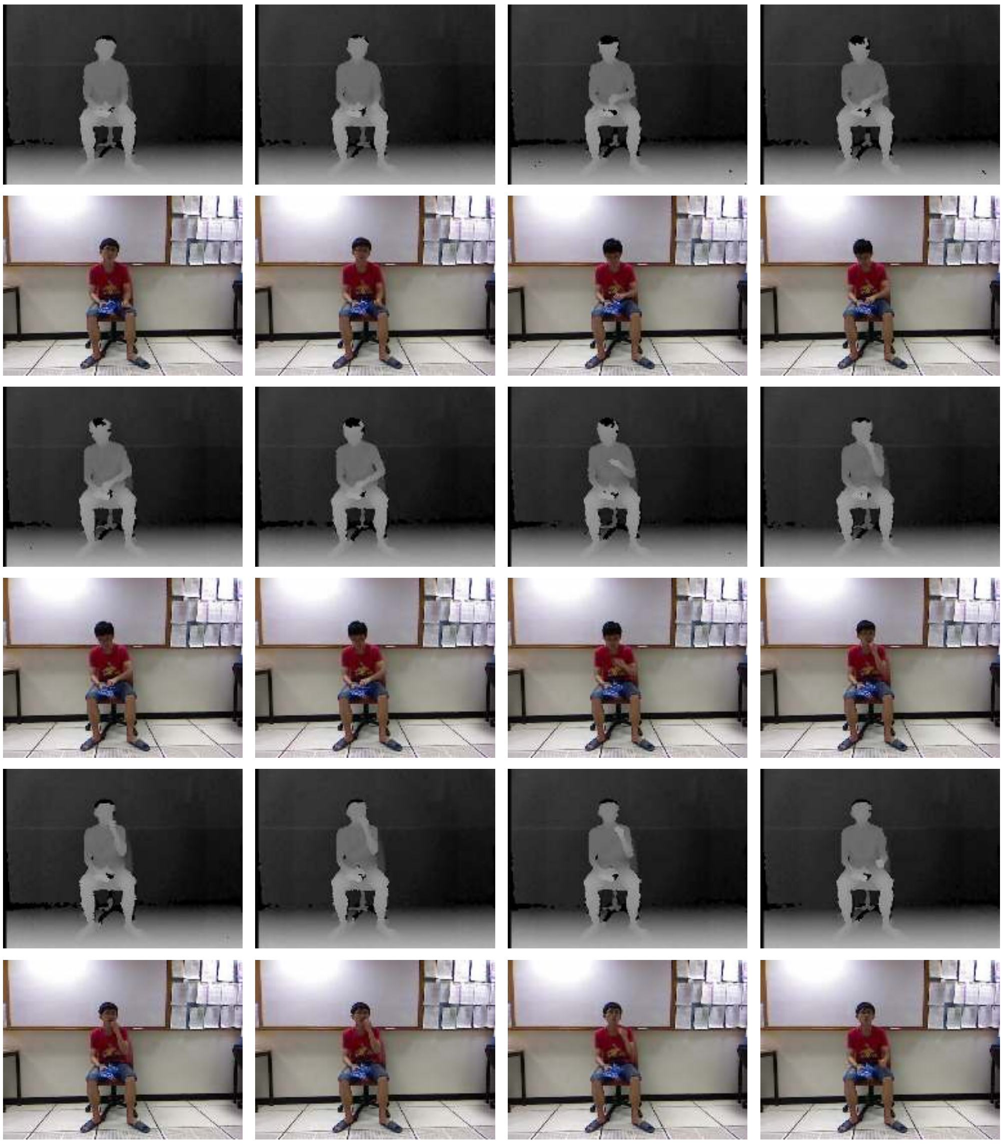


Fig. 2.16 Illustration of the depth-maps and corresponding RGB frames from a sample in Online RGB+D Action dataset. This sample shows the action “eating”.

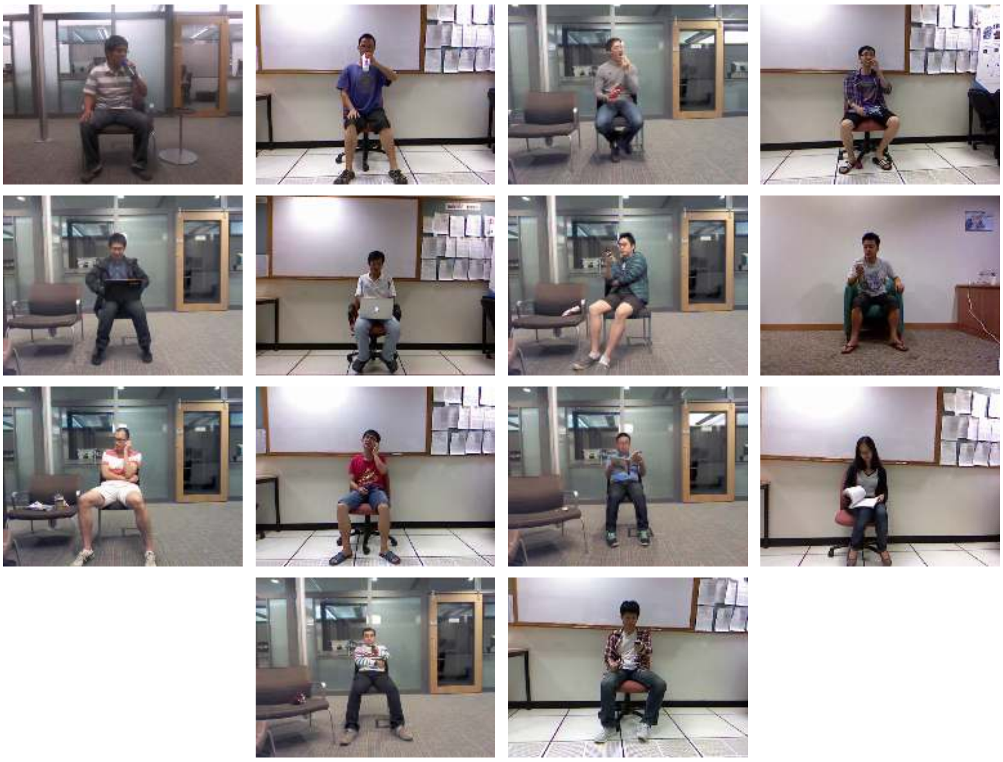


Fig. 2.17 RGB sample frames of 7 different actions in Online RGB+D Action dataset. Actions are performed in two different environments.

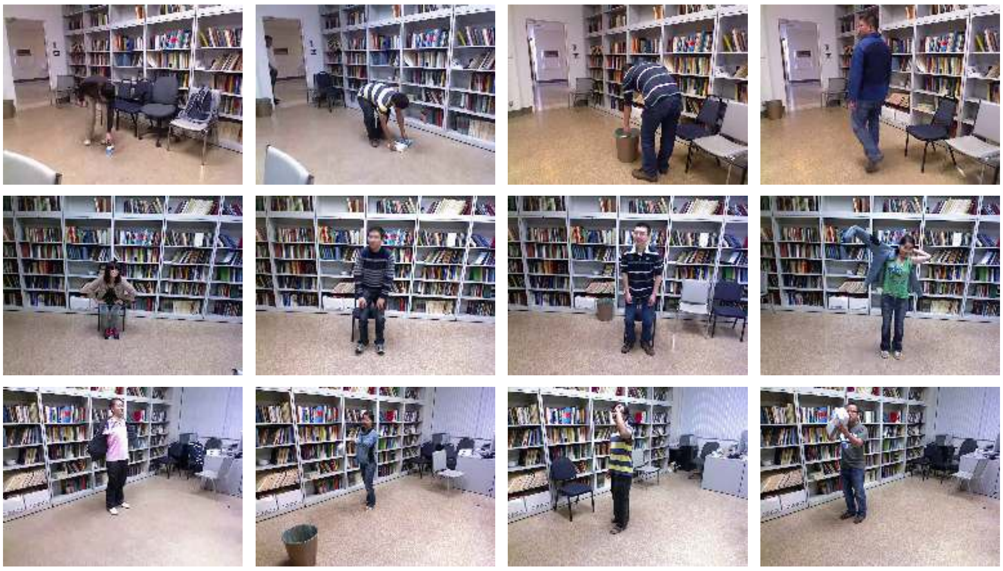


Fig. 2.18 RGB sample frames of different action classes in Northwestern-UCLA dataset. The samples are taken in 3 different views.

Cross-View Protocol

In multi-view datasets, cross-view evaluation protocol suggests to use some of the camera views for training and keep unseen views for testing to evaluate the robustness of the recognition algorithms in new view-points.

Cross-Environment Protocol

For the datasets which have samples captured in different environment, cross-environment protocol is another option for evaluation to verify the strengths of the methods in different background conditions.

Leave-One-Subject-Out Cross-Validation

A more sophisticated evaluation protocol is repeat the experiment once per each subject and on each round keep that subject out for testing and use all others for training. The final performance will be calculated as the average of the performances over all the iterations.

Datasets	Samples	Classes	Subjects	Views	Sensor	RGB	Depth	Joints	Other	Year
MSR-Action3D [51]	567	20	10	1	N/A		✓	✓		2010
CAD-60 [90]	60	12	4	-	Kinect 360	✓	✓	✓		2011
RGBD-HuDaAct [63]	1189	13	30	1	Kinect 360	✓	✓			2011
MSRDailyActivity3D [109]	320	16	10	1	Kinect 360	✓	✓	✓		2012
ACT4^2 [18]	6844	14	24	4	Kinect 360	✓	✓			2012
UTKincet [121]	200	10	10	-	Kincet 360	✓	✓	✓		2012
SBU [131]	300	8	7	1	Kinect 360	✓	✓	✓		2012
CAD-120 [39]	120	20	4	-	Kinect 360	✓	✓	✓		2013
3D Action Pairs [67]	360	12	10	1	Kinect 360	✓	✓	✓		2013
Multiview 3D Event [118]	3815	8	8	3	Kinect 360	✓	✓	✓		2013
Online RGB+D Action [128]	336	7	24	1	Kinect 360	✓	✓	✓		2014
G3Di [7]	72	6	12	1	Kinect 360	✓	✓	✓		2014
LIRIS Human Activ. [119]	828	10	21	-	Kinect 360	✓	✓			2014
Office Activity [114]	1180	20	5	-	Kinect 360	✓	✓			2014
Northwestern-UCLA [111]	1475	10	10	3	Kinect 360	✓	✓	✓		2014
UWA3D Multiview [73]	~900	30	10	1	Kinect 360	✓	✓	✓		2014
Office Activity [115]	1180	20	10	3	Kinect 360	✓	✓			2014
UTD-MHAD [16]	861	27	8	1	Kinect 360+WIS	✓	✓	✓	ID	2015
UWA3D Multiview II [71]	1075	30	10	5	Kinect 360	✓	✓	✓		2015

Table 2.1 A comparison between some of the currently available datasets for depth-based action recognition.

Chapter 3

Multimodal Multipart Learning for Action Recognition in Depth Videos

In our first work, the focus is on the 3D representations of the human bodies and their kinetics in the context of activity analysis. We study the depth-based and 3D skeleton-based features and propose a new framework to perform the feature analysis and action classification effectively.

3.1 Introduction

Human actions consist of simultaneous flow of different body parts. Based on this complex articulated essence of human movements, the analysis of these signals could be highly complicated. To ease the task of classification, actions could be broken down into their components. This is done by a body part detection on depth sequences of human body movements [85]. Having the 3D locations of body joints in the scene, we can separate the complicated motion of body into a concurrent set of behaviors on major skeleton joints; therefore human action sequences can be considered as multipart signals. Throughout this chapter, we use the term “part” to denote each body joint as defined in [85].

Limiting the learning into skeleton based features cannot deliver high levels of performance in action recognition, because: (1) most of the usual human actions are defined based on the interaction of body with other objects, and (2) depth based skeleton data is not always accurate due to the noise and occlusion of body parts. To alleviate these issues, different depth based appearance features can be leveraged. The work in [110] proposed LOP (local occupancy patterns) around each of the body joints in order to represent 3D appearance of the interacting objects. Another solution is HON4D (histogram of oriented 4D normals) [67], which gives more descriptive and robust models of the local depth based appearance and motion, around the joints. Based on the complementary properties of mentioned features, it is beneficial to utilize all of them as different descriptors for each joint. Combining heterogeneous features of each part of the skeleton, leads into a multimodal-multipart combination, which demands sophisticated fusion algorithms.

An interesting approach to handle the articulation of actions was recently proposed by [110]. As the key intuition, they have shown each individual action class can be represented by the behavior and appearance of few informative joints in the body. They utilized a data mining technique to find these discriminative sets of joints for each class of the available actions and tied up the features of those parts as "actionlets". They employed a multi-kernel learning method to build up ensembles of actionlets as kernels for action classification. This method is highly robust against the noise in depth maps, and the results show its strength to characterize human body motions and also human-object interactions. However the downside of this approach is the inconsistency of their heuristic selection process (mining actionlets) with the following learning step. Moreover, it simply concatenates different types of features for multimodal fusion, which is another drawback of their work. In this fashion, achieving the optimal combination of features regarding the classification task cannot be guaranteed.

To overcome the limitations mentioned above, we propose a joint structured sparsity regression based learning method which integrates part selection into the learning process considering the heterogeneity of features for each joint. We associate all the features for each part as a bundle and apply a group sparsity regularization to select a small number of active parts for each action class. To model the precise hierarchy of the multimodal-multipart features in an integrated learning and selection framework, we propose a hierarchical mixed norm which includes three levels of regularization over the learning weights. To apply the modality based coupling over heterogeneous features of each part, it applies a mixed norm with two degrees of "diversity" induction [78], followed by a group sparsity among the feature groups of different parts to apply part selection.

Mixed norms are powerful tools to inject simultaneous sparsity and coupling effects between the learning coefficients. They have been studied in a variety of fields. In statistical domain, Yuan and Lin [129] proposed the "group Lasso", as an extension over "Lasso" [93] for a grouped variable selection in regression. Zhao *et al.* [135] introduced "composite absolute penalty" for hierarchical variable selection. "Hierarchical penalization" is also proposed to utilize prior structure of the variables for a better fitting model [92]. In sparse regression, mixed norms have been used as regularization terms to link sparsity and persistence of variables [41]. A generalized shrinkage scheme was proposed by [42] for structured sparse regression. Wang *et al.* [106] used mixed norms as structured sparsity regularizers for heterogeneous feature fusion, and [104] extended this idea for multi-view clustering. They proposed robust self-taught learning [105] using mixed norms and [103] utilized a fractional mixed norm for robust adaptive dictionary learning. In the proposed work in this chapter, we regularize the multimodal features of each part by applying a mixed L^2/L^4 norm. To achieve the sparsity between parts, we generalize this to an $L^1/L^2/L^4$ hierarchical norm.

If multiple learning tasks at hand share some inherent constituents or structures, "Multitask Learning" [14, 138, 116, 1] techniques could be globally beneficial. In joint sparse regression, multitask learning is formulated by a mixed norm. Liu *et al.* [52] proposed an L^1/L^∞ norm to add this into Lasso for variable selection. In joint feature selection, L^1/L^2 norm can provide multitask learning by applying selection between the L^2 regularized parameters of each feature [64]. The same is used in [130] as a generalization of L^1 norm in a multitask joint sparsity representation model to fuse complementary visual features across recognition tasks. Zhang *et al.* [133] studied different mixed norms when they applied multitask sparse learning in visual tracking and based on their experimental results, they showed that L^1/L^2 is superior among them. In this work, we use a similar norm to utilize the shared latent factors between different binary action classifiers. We apply L^2 regularization over the weights corresponding to each feature across all the tasks, followed by an L^1 between all the features at hand.

We evaluate our method on four challenging depth-based action recognition datasets: MSR Daily Activity 3D dataset [110], MSR Action3D dataset [51], 3D ActionPairs dataset [67], and NTU RGB+D Action dataset (section 5.2 and [81]). Our experimental results show that the proposed method is superior to other available methods for action recognition on depth sequences.

The rest of this chapter is organized as follows: Section 3.2 presents the proposed integrated feature selection and learning scheme. It also introduces the new multimodal-multi part mixed norm which applies regularization and group sparsity into the proposed learning model. Experimental results on three above-mentioned benchmarks are covered in section 3.3 and we conclude the chapter in section 3.4.

3.2 Multimodal Multipart Learning

Notations

Throughout this chapter, we use bold uppercase letters to represent matrices and bold lowercase letters to indicate vectors. For a matrix $\mathbf{X}$, we denote its j-th row as $\mathbf{x}^j$ and its i-th column as $\mathbf{x}_i$.

Assume the partition ξ is defined over a vector $\mathbf{z}$ to divide its elements into $|\xi|$ disjoint sets. We use ξ_i to represent the indices of the i-th set in ξ, and its corresponding elements in $\mathbf{z}$ are referred to as $\mathbf{z}^{\xi_\mathbf{i}}$, also $z^{\xi_\mathbf{i},k}$ represents the k-th element of $\mathbf{z}^{\xi_\mathbf{i}}$. The L^p/L^q norm of $\mathbf{z}$ regarding ξ is represented by $\|\mathbf{z}\|_{q,p|\xi}$ and is defined as the L^q norms of the elements inside each set of ξ followed by an L^p norm of the L^q values across the sets; mathematically:

$$\|\mathbf{z}\|_{q,p|\xi} = \left(\sum_{i=1}^{|\xi|} \|\mathbf{z}^{\xi_\mathbf{i}}\|_q^p\right)^{1/p} = \left(\sum_{i=1}^{|\xi|}\left(\sum_{k=1}^{|\xi_i|} |z^{\xi_\mathbf{i},k}|^q\right)^{p/q}\right)^{1/p} \tag{3.1}$$

in which $|\xi_i|$ indicates the cardinality of set ξ_i.

Now consider the elements of each set ξ_i are further partitioned by operator ρ into $|\rho|$ disjoint subsets. Similarly, we indicate j-th ρ-subset of i-th ξ-set of $\mathbf{z}$ as $\mathbf{z}^{\xi_\mathbf{i},\rho_\mathbf{j}}$ and $z^{\xi_\mathbf{i},\rho_\mathbf{j},k}$ represents its k-th element. The $L^p/L^q/L^r$ norm of $\mathbf{z}$ regarding ξ and ρ is also represented by $\|\mathbf{z}\|_{r,q,p|\rho,\xi}$ and is defined as the L^q/L^r norms (regarding ρ) of all $|\xi|$ sets followed by an L^p norm of the L^q/L^r values across the sets of ξ; mathematically:

$$\|\mathbf{z}\|_{r,q,p|\rho,\xi} = \left(\sum_{i=1}^{|\xi|} \|\mathbf{z}^{\xi_\mathbf{i}}\|_{r,q|\rho}^p\right)^{1/p} = \left(\sum_{i=1}^{|\xi|}\left(\sum_{j=1}^{|\rho|}\left(\sum_{k=1}^{|\rho_j|} |z^{\xi_\mathbf{i},\rho_\mathbf{j},k}|^r\right)^{q/r}\right)^{p/q}\right)^{1/p} \tag{3.2}$$

This representation can be easily extended into higher orders of structural mixed norms by further partitioning the subsets.

3.2.1 Multipart Learning by Structured Sparsity

Our purpose of learning is to recognize the actions in depth videos, based on depth and skeleton based features extracted. The set of input features we use to describe each action sample is a combination of multimodal multipart features. The entire body is separated into a number of parts (as illustrated in Fig.3.1) and for each part we have different types of features to represent the movement and local depth appearance. In

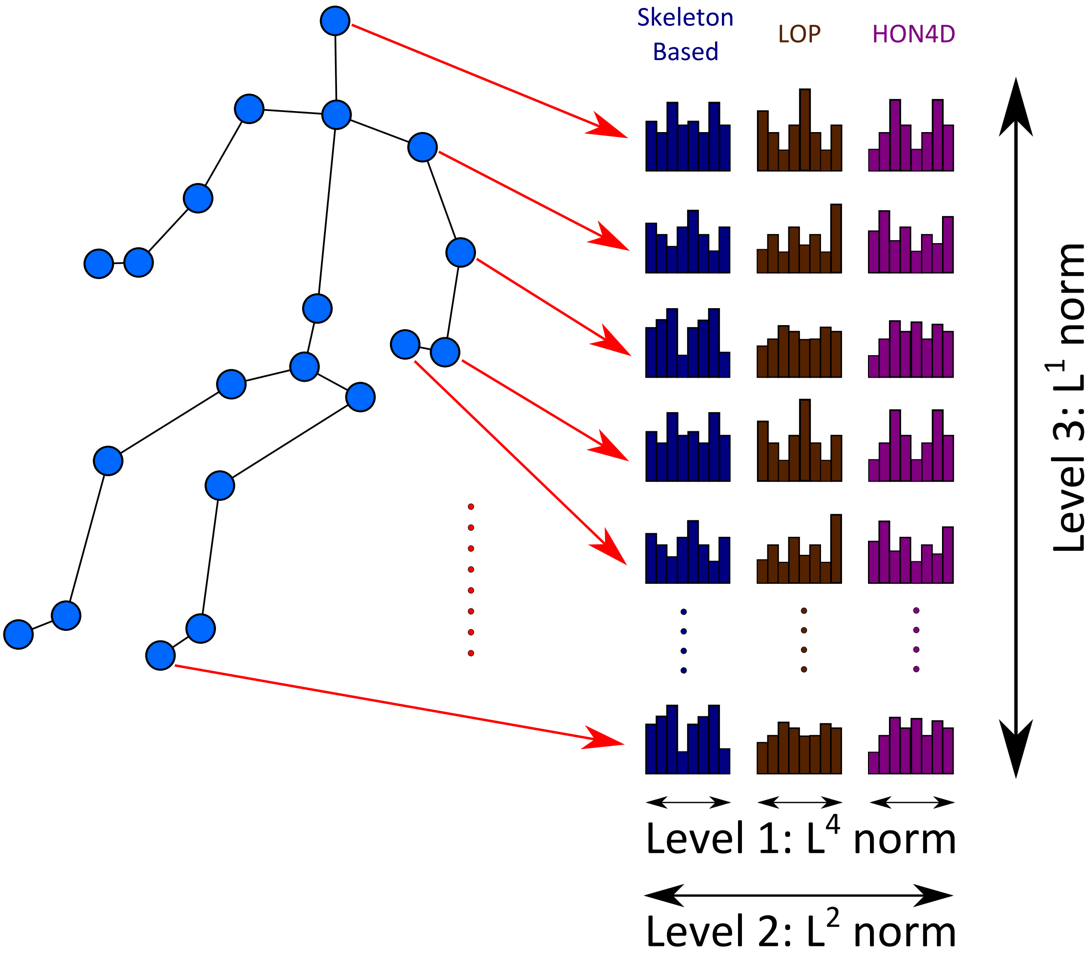


Fig. 3.1 Three Levels of the Proposed Hierarchical Mixed Norm for Multimodal Multipart Learning. We combine two levels of regularization inside modality groups and between them for each part, followed by a sparsity inducing norm between the parts to apply part selection.

this work, we consider each body joint as a part of the body and our multipart analysis is performed among the joints.

Therefore, our input feature set for each input sample, can be represented by a vector: $\mathbf{z} \in \mathbb{R}^d$, which consists of feature groups of different parts and modalities. Assume operator π is partitioning $\mathbf{z}$ into P parts, and μ is defined over sets of π to further partition them based on M number of feature modalities. So, the hierarchy of features inside this vector is indicated by: $\mathbf{z} = [\mathbf{z}^{\pi_{\mathbf{1}}T}, \dots, \mathbf{z}^{\pi_{\mathbf{P}}T}]^T$, in which each $\mathbf{z}^{\pi_{\mathbf{i}}} = [\mathbf{z}^{\mu_{\mathbf{1}},\pi_{\mathbf{i}}T}, \dots, \mathbf{z}^{\mu_{\mathbf{M}},\pi_{\mathbf{i}}T}]^T$.

Now the problem of multiclass action recognition can be considered as multiple binary regression based classification problems in a one-versus-all manner. Given n training samples $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]$ in which $\mathbf{x}_i \in \mathbb{R}^d$ and their corresponding labels for C distinct classes: $\mathbf{Y} = [\mathbf{y}_1, ..., \mathbf{y}_C]$ with $\mathbf{y}_c \in \{0,1\}^n$ and $\forall i : \sum_{c=1}^{C} y_c^i = 1$; we are looking for a projection matrix $\mathbf{W}^* \in \mathbb{R}^{d \times C}$ which minimizes a set of loss functions $J^c(\langle \mathbf{x}_i, \mathbf{w}_c^* \rangle, y_c^i)$ for all classes $c \in \{1, ..., C\}$ and samples $i \in \{1, ..., n\}$. Our choice for the total loss function, without loss of generality, is sum of squared errors ($\forall c :$ $J^c(a,b) = (a-b)^2$).

The most common shrinkage methods to regularize the learning weights against overfitting are to penalize L^p norms of the learning weights for each class:

$$\mathbf{w}_c^* = \underset{\mathbf{w}_c}{\operatorname{argmin}} \sum_{i=1}^{n} J^c(\langle \mathbf{x}_i, \mathbf{w}_c \rangle, y_c^i) + \lambda \|\mathbf{w}_c\|_p \tag{3.3}$$

in which λ is the regularization factor. Employing L^2 norm ($p = 2$) leads into a general weight decay and minimization of the magnitude of $\mathbf{W}$, and applying L^1 norm ($p = 1$) yields simultaneous shrinkage and sparsity among the individual features. Such methods simply ignore the structural information between the features, which can be useful for classification; therefore, it is beneficial to embed these feature relations into our learning scheme via structured sparsity inducing mixed norms.

In the context of depth based action recognition, features are naturally partitioned into parts. "Actionlet ensemble" method [110] tried to discover discriminative joint groups using a data mining process, which led into an interesting improvement on the performance; however, their heuristic selection process is discrete and separated from the following learning step. To address these issues, we propose to apply group sparsity to perform part selection and classification in a regression based framework, in contrast to the mining based joint group discovery of [110].

We know that the discriminative strength of features in each part are highly correlated regarding all the classes at hand. So we expect the corresponding learning parameters (elements of each $\mathbf{w}_c$) to be triggered or halted concurrently within each set of π partitioning (for each action class). To apply a grouping effect on these features, we consider each set in π as a unit and measure its strength with an L^2 norm of the included learning weights. On the other hand, we seek a sparse set of parts to be activated for each class at hand, so we apply an L^1 norm between the L^2 values of the groups. Such an intuition can be formulated by an L^1/L^2 mixed norm based on π for each class:

$$\mathbf{w}_c^* = \underset{\mathbf{w}_c}{\operatorname{argmin}} \sum_{i=1}^{n} J^c(\langle \mathbf{x}_i, \mathbf{w}_c \rangle, y_c^i) + \lambda \|\mathbf{w}_c\|_{2,1|\pi} \tag{3.4}$$

Adding this up for all the action classes with the same regularization factor, we have:

$$\mathbf{W}^* = \underset{\mathbf{W}}{\operatorname{argmin}} \sum_{c=1}^{C} \sum_{i=1}^{n} J^c(\langle \mathbf{x}_i, \mathbf{w}_c \rangle, y_c^i) + \lambda \sum_{c=1}^{C} \|\mathbf{w}_c\|_{2,1|\pi} \quad , \tag{3.5}$$

$$= \underset{\mathbf{W}}{\operatorname{argmin}} \mathbf{J}(\mathbf{X}^T\mathbf{W}, \mathbf{Y}) + \lambda \|vec(\mathbf{W})\|_{2,1,1|\pi,\tau} \quad , \tag{3.6}$$

in which $vec(.)$ is the vectorization operator and τ is the partitioning operator of $vec(\mathbf{W})$ elements based on their corresponding tasks (or columns here): $\forall(k,c): \ \tau(w_c^k) = c$. We will refer to this multipart learning method as “MP”.

Minimization of (3.6) applies the desired grouping effect into the features of each part and guarantees the sparsity on the number of active parts for each class in a smooth and simpler way, compared to the actionlet method.

3.2.2 Multimodal Multipart Learning via Hierarchical Mixed Norm

In the above formulation, we apply an L^2 regularization norm over heterogeneous features of all the modalities for each part, and ignore the modality structures between them. In other words, applying a general L^2 norm may cause the suppression of the information at some dimensions. These issues are more severe when training samples are limited (which is the case for action recognition in depth), in which it might lead to weak generalization of the learning.

To overcome these limitations, we utilize L^∞ to regularize the coefficients inside each modality, so that “diversity” [41] can be encouraged. It is already known that the behavior of L^p norm for $p > 2$ rapidly moves towards L^∞ [76]; since L^∞ is not easy to optimize directly, we picked L^4 as the most efficient approximation of it. Higher order norms like L^6 apply the same effect but with a slightly more expensive processing cost.

By applying the L^4 norm to regularize the weights in each modality group of each part, now we have a three-level $L^1/L^2/L^4$ mixed norm. Inner L^4 gives more “diversity” to regularize the features inside each partiality-modality subset. L^2 norm employs a magnitude based regularization over the L^4 values to link different modalities of each part, and the outer L^1 applies the soft part selection between the L^2/L^4 values of each action class (Fig.3.1).

Replacing the previous structured norm by the proposed hierarchical mixed norm in (3.6), we have:

$$\mathbf{W}^* = \underset{\mathbf{W}}{\operatorname{argmin}} \mathbf{J}(\mathbf{X}^T\mathbf{W},\mathbf{Y}) + \lambda \sum_{c=1}^{C} \|\mathbf{w}_c\|_{4,2,1|\mu,\pi} \tag{3.7}$$

$$= \underset{\mathbf{W}}{\operatorname{argmin}} \mathbf{J}(\mathbf{X}^T\mathbf{W},\mathbf{Y}) + \lambda \|vec(\mathbf{W})\|_{4,2,1,1|\mu,\pi,\tau} \tag{3.8}$$

Here, π indicates the partitioning of features based on their source body part, and μ represents further partitioning of each part's set regarding the modalities of the features. Table 3.1 lists all the partitioning operators used in this work. In the rest of this chapter, we use the abbreviation "MMMP" to refer to this method. It is worthwhile to note that changing the inner norm to L^2 will reduce the hierarchical norm into a two level mixed norm, i.e. $\|vec(\mathbf{W})\|_{2,2,1,1|\mu,\pi,\tau} = \|vec(\mathbf{W})\|_{2,1,1|\pi,\tau}$ derived directly from the definition of hierarchical norm (3.2).

When different learning tasks have similar latent features, "Multitask Learning" [14] techniques can improve the performance of the entire system by applying information sharing between the tasks. Here we are learning classifiers for C different classes which essentially have many latent components in common, so pushing them to share some features is beneficial for the classification task. This can be done by applying an L^2 grouping on all the weights corresponding to each individual feature. Each of these L^2 values represents the magnitude of strength for its corresponding feature among all the tasks. Then applying an L^1 over the magnitudes can apply a shared variable selection considering all the tasks. Adding the new multitask term into (3.8), we have:

$$\mathbf{W}^* = \underset{\mathbf{W}}{\operatorname{argmin}} \mathbf{J}(\mathbf{X}^T\mathbf{W},\mathbf{Y}) + \lambda_1 \sum_{k=1}^{d} \|\mathbf{w}^k\|_2 + \lambda_2 \|vec(\mathbf{W})\|_{4,2,1,1|\mu,\pi,\tau} \tag{3.9}$$

$$= \underset{\mathbf{W}}{\operatorname{argmin}} \mathbf{J}(\mathbf{X}^T\mathbf{W},\mathbf{Y}) + \lambda_1 \|vec(\mathbf{W})\|_{2,1|\phi} + \lambda_2 \|vec(\mathbf{W})\|_{4,2,1,1|\mu,\pi,\tau} \tag{3.10}$$

Here, d is the number of rows in $\mathbf{W}$ which is equal to the size of the entire feature vector, and ϕ defines the partitioning of $vec(\mathbf{W})$ elements based on their corresponding individual features: $\forall(k,c): \phi(w_c^k) = k$. For a review on the used partitioning operators in this work, please see Table 3.1.

Combining these two regularization terms can be considered as a trade off between sparsity and persistence of features [43] based on their relations across the parts, modalities, and between the action classes.

In our experiments, we use $P = 20$ body joints as partitioning operator π. Since each column of $\mathbf{W}$ has the same hierarchical partitioning as input features: $\mathbf{W} = [\mathbf{w}_c^j]$,

Table 3.1 Definition of the different partitioning operators in the proposed method.

Symbol	Name	Definition
π	part-based	partitioning based on the index of the corresponding body joint
μ	modality-based	partitioning based on the index of the feature group
τ	task-based	partitioning based on the index of the corresponding action class or the index of the columns in the weight matrix
ϕ	individual feature-based	partitioning based on the index of each feature element or the index of the rows in the weight matrix

in which c counts the number of classes and j counts the feature groups for P joints. The features for each joint come from $\mathbf{M} = 3$ different modalities: skeletons, LOP, and HON4D; this defines the μ operator. Therefore, each $\mathbf{w}_c^j = [\mathbf{w}_c^{j,1^T}, ..., \mathbf{w}_c^{j,M^T}]^T$, in which each $\mathbf{w}_c^{j,m}$ is the corresponding weight elements to class c, joint j and modality m. This way (3.10) will be expanded to:

$$\mathbf{W}^* = \underset{\mathbf{W}}{\operatorname{argmin}} \|\mathbf{X}^T\mathbf{W} - \mathbf{Y}\|_F^2 + \lambda_1 \sum_{k=1}^{d} \|\mathbf{w}^k\|_2 + \lambda_2 \sum_{c=1}^{C} \sum_{j=1}^{P} (\sum_{m=1}^{M} \|\mathbf{w}_c^{j,m}\|_4^2)^{1/2} \qquad (3.11)$$

3.2.3 Two Step Learning Approach

The size of the learning weights are very large in comparison to the size of training samples, specifically for older benchmark datasets. This limits the ability of a one-step overall learning framework to converge to an ideal optimum point. To resolve this difficulty, we break this to two easier-to-solve steps. First we learn the partially optimum weights for multipart features of each modality separately. In the second step, we solve the overall multimodal multipart optimization problem by fine-tuning the weights and prevent them from deviating from their corresponding partially optimum points.

Since each of the modalities has the part-based structure in their input features, we use the proposed multipart learning in presence of multitask learning term. To learn the partially optimum weights for input modality m, we optimize:

$$\widehat{\mathbf{W}_\mathbf{m}} = \underset{\mathbf{W_m}}{\operatorname{argmin}} \mathbf{J}(\mathbf{X}_\mathbf{m}^T\mathbf{W_m}, \mathbf{Y}) + \hat{\lambda}_1 \|vec(\mathbf{W_m})\|_{2,1|\phi} + \hat{\lambda}_2 \|vec(\mathbf{W_m})\|_{2,1,1|\pi,\tau} \quad (3.12)$$

where $\mathbf{X_m}$ denotes the input features for m^th modality.

After achieving the partially optimum point for each modality, we merge the $\widehat{\mathbf{W}_\mathbf{m}}$ values for all M modalities:

$$\widehat{\mathbf{W}} = [\widehat{\mathbf{W}}_1^T, ..., \widehat{\mathbf{W}}_M^T]^T \quad (3.13)$$

Next is to fine-tune the weights in the multimodal-multipart learning fashion, on a neighborhood of $\widehat{\mathbf{W}}$ values. To do so, we expect the global optimum weight not to diverge too much from their partially optimal points:

$$\begin{aligned} \mathbf{W}^* = \underset{\mathbf{W}}{\operatorname{argmin}} \mathbf{J}(\mathbf{X}^T\mathbf{W}, \mathbf{Y}) + \lambda_1 \|vec(\mathbf{W})\|_{2,1|\phi} \\ + \lambda_2 \|vec(\mathbf{W})\|_{4,2,1,1|\mu,\pi,\tau} + \lambda_3 \|\mathbf{W} - \widehat{\mathbf{W}}\|_F^2 \end{aligned} \quad (3.14)$$

The last term in (3.14) will limit the deviation of learning weights from their partially optimal point, as we expect them to be just fine-tuned in this step.

Upon optimization over training data, the detection of the learned classifier for each testing sample $\mathbf{x}_i$ can be obtained by:

$$f(\mathbf{x}_i) = \underset{c}{\operatorname{argmax}} \langle \mathbf{x}_i, \mathbf{w}_c^* \rangle \quad (3.15)$$

Broyden–Fletcher–Goldfarb–Shanno (BFGS) is an optimization algorithm that solves the unconstrained minimization problem iteratively. All it needs at each iteration is the value of the cost function and its gradient. This algorithm does not require the Hessian matrix of the cost function, neither its inverses. The limited-memory BFGS (L-BFGS) iteratively finds a minimizer by approximating the inverse of the Hessian matrix of the cost function by utilizing the information from last iterations. This technique saves the memory usage and computational complexity for large-sized problems.

Our optimization steps are all done by Limited memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) algorithm using off-the-shelf “minFunc” tool [80].

Another limitation of the proposed work in this chapter is the batch nature of the optimization algorithm. All the input features and learning weights are supposed to be kept in memory throughout the learning process. This limits the ability of this method to combine a big number of features or learn on very large scale datasets.

3.3 Experiments

This section describes our experimental setup details and then provides the results of the proposed method on three depth based action recognition benchmarks. The description of the datasets are provided in section 2.3.1.

3.3.1 Experimental Setup

All the provided experiments are done on Kinect based datasets. Kinect captures RGB frames, depth map signals and 3D locations of major joints. To have a fair comparison with other depth based methods, we ignore the RGB signals. Skeleton extraction is done automatically by Kinect's SDK based on the part-based human pose recognition system of [85]. On each frame, we have an estimation of 3D positions of 20 joints in the body. All of our features are defined based on these joints as the multipart partitioning operator (π); therefore, each feature necessarily belongs to one of these parts.

To represent skeleton based features, first we normalize the 3D locations of joints against size, position and direction of the body in the scene. The original body joints data are measured in camera coordinate system. First we translate all of the joints to the body coordinate system. The origin of the new coordinate system is on the "middle of the spine" joint (number 3 in Fig. 1.3), followed by a 3D rotation to fix the X axis towards the vector from "right shoulder" to "left shoulder", and Y axis from "spine base" to "spine". The new Z axis is obtained from $X \times Y$. In the last step of normalization, we scale all the joint point locations based on the distance between "spine base" and "spine" joints. This normalization step eases the task of comparison between body poses. On the other hand, it removes the location and rotation information of the body at each frame. This extracted body locations and directions could also be highly discriminative for some action classes like "walking" or "lying down"; therefore we add the translation and rotation vectors of the normalization step into the features under a new auxiliary part.

To encode the dynamics of skeleton based features, we apply "Fourier temporal pyramid" as suggested by [110] and keep first four frequency coefficients of each short time Fourier transformation. This leads to a feature vector of size 1,876 for each action sample.

In addition to skeleton based features, other modalities we use are local HON4D [67] and LOP [110] to represent depth based local dynamics and appearance around each joint. On each frame, LOPs are extracted on a (96,96,320)-sized depth neighborhood of each joint, which is divided into $3 \times 3 \times 4$ number of (32,32,80)-sized bins. To represent LOP based kinetics, we use a similar Fourier temporal pyramid transformation. HON4D features are also extracted locally over the location of joints on each frame. We encode HON4D features using LLC (locality-constrained linear coding) [113] to reduce their dimensionality while preserving the locality of 4D surface normals. A dictionary size of 100 is picked for the clustering step. LLC codes go through a max pooling over a 3 level temporal pyramid. Dimensions of the features for LOP and HON4D are 5,040 and 14,000 respectively. The overall dimensionality of input features for each sample is 20,916.

The hyper-parameters $\widehat{\lambda_1}, \widehat{\lambda_2}, \lambda_1, \lambda_2$, and λ_3 are set by leave-one-sample-out cross-validation over training data. We found the optimal values for these hyper-parameters between $[1e^{-5}, 1e^{-4}, ..., 1, ...1e^{+5}]$ values.

3.3.2 MSR Daily Activity 3D Dataset

First, to verify the strengths of our proposed hierarchical mixed norm, we evaluate the performance of the classification in a subject-wise cross-validation scenario. We evaluate the performance of the plain L^2 norm, the multipart structured norm (MP), and the proposed hierarchical mixed norm (MMMP), in all 252 possible train/test splits of 5 out of 10 subjects. To have a proper comparison between these norms, we have not applied the multitask term. The results of this experiment are shown in Table 3.2. Adding part based grouping, when it ignores the modality associations between the features, can slightly improve the performance from 80.61% into 81.55%. By adding multimodality grouping and applying the proposed hierarchical mixed norm, improvement is more significant and reaches 84.03%.

Next, we verify the results of our method by applying mentioned norms on the standard train/test split of the subjects. As provided in Table 3.3, applying simple feature selection using a plain L^1 norm leads into 86.88% of accuracy. By applying a plain L^2 norm on all the features we get 87.50%. Multipart learning regardless of heterogeneity of the modalities leads into 88.13%. Finally by adding the multipart learning via the proposed hierarchical mixed norm we reach the interesting accuracy of 91.25% on this dataset. Applying higher orders for the inner-most norm (like $L^1/L^2/L^6$) achieved the same level of accuracy at a slightly higher processing time.

Table 3.2 Subject-wise Cross-Validation Performance Comparison of the Proposed Hierarchical Mixed Norm with Plain and Multipart Group Sparsity Norm on the MSR Daily Activity 3D Dataset

Method	**Structure/Hierarchical Norm Used**	**Accuracy**
L^2	$\|\|vec(\mathbf{W})\|\|_2^2$	80.61±2.49%
MP	$\|\|vec(\mathbf{W})\|\|_{2,1,1\|\pi,\tau}$	81.55±2.43%
MMMP	$\|\|vec(\mathbf{W})\|\|_{4,2,1,1\|\mu,\pi,\tau}$	84.03±2.16%

To assess the strength of the proposed multipart learning, we evaluate our method on single modality setting using (3.12). As shown in Table 3.4, on skeleton based features, we got 79.38% compared to 74% of the baseline actionlet method. Using LOPs, our method achieved 79.38% which is more than 18 percentage points higher than the actionlet's performance. For local HON4D features, we achieved 81.88% compared to 80.00% of the baseline local HON4D method. Now we use the partially learned weights of single modality multipart learning and employ them for the optimization of (3.14) to learn globally optimum projections. First we try the combination of skeleton based features with LOP. Using proposed learning, we get 88.13% of accuracy which outperforms the baseline's best result of 85.75%. [112] used skeleton and HON4D features in a temporal warping framework and got 88.75%. Our method outperforms it using the same set of features by achieving 89.38% of accuracy. And finally using all three modalities, our method achieves the performance level of 91.25%. Table 3.4 shows the complete set of results for this experiment.

Our implementation is done in MATLAB, and not fully optimized for time efficiency. The average training and testing time of MMMP on a 3.2 *GHz* Core-i5 machine are 170 and 2×10^{-4} seconds respectively, with no parallel processing.

It is worth pointing out some of the published works on this dataset applied other train/test splits, e.g. [3] reported 93.1% of accuracy on a leave-one-subject-out cross-validation. On this setup, proposed MMMP method achieves 97.5%.

3.3.3 MSR Action 3D Dataset

The reported results on this dataset are divided in two different scenarios. First is the average cross-subject performance on three action subsets defined in [51], and second is the overall cross-subject accuracy regardless of subsets, as done in [110]. Following [100], we call them as protocols of [51] and [110].

Table 3.3 Performance Comparison of the Proposed Method Using Plain/ Structured/ Hierarchical Norms on the Standard Evaluation Split of the MSR Daily Activity 3D Dataset

Method	**Structure/Hierarchical Norm Used**	**Accuracy**
L^1	$\|vec(\mathbf{W})\|_1$	86.88%
L^2	$\|vec(\mathbf{W})\|_2^2$	87.50%
MP	$\|vec(\mathbf{W})\|_{2,1,1\|\pi,\tau}$	88.13%
MMMP	$\|vec(\mathbf{W})\|_{4,2,1,1\|\mu,\pi,\tau}$	91.25%

Table 3.4 Performance Comparison on the Standard Evaluation Split of the MSR Daily Activity 3D Dataset using Single Modality and Multimodal Features.

Method	**Modalities**	**Accuracy**
Actionlet Ensemble [110]	LOP	61%
Proposed MP	LOP	**79.38%**
Orderlet Mining [128]	Skeleton	73.8%
Actionlet Ensemble [110]	Skeleton	74%
Proposed MP	Skeleton	**79.38%**
Local HON4D [67]	HON4D	80.00%
Proposed MP	HON4D	**81.88%**
Actionlet Ensemble [110]	Skeleton+LOP	85.75%
Proposed MMMP	Skeleton+LOP	**88.13%**
MMTW [112]	Skeleton+HON4D	88.75%
Proposed MMMP	Skeleton+HON4D	**89.38%**
DSTIP [120]	DCSF+LOP	88.20%
Proposed MMMP	Skeleton+LOP+HON4D	**91.25%**

Tables 3.5 and 3.6 show the results. Although we still have the highest accuracy among the reported results, the achieved margin is not as large as other datasets. This is because of the simplicity of actions in this dataset. Since there are no interactions with other objects, most of the classes are highly distinguishable using skeleton only features; therefore our multimodality could not boost up the results that much, but the multipart learning still shows its advantage over other methods.

Table 3.5 Average Cross-Subject Performance for MSR Action 3D Dataset on Three Action Subsets of [51]

Method (protocol of [51])	Accuracy
Action Graph on Bag of 3D Points [51]	74.7%
Histogram of 3D Joints [121]	79.0%
EigenJoints [124]	83.3%
Random Occupancy Patterns [108]	86.5%
Skeletal Quads [24]	89.9%
HOG3D-LLC [70]	90.9%
Depth HOG [126]	91.6%
Lie Group [100]	92.5%
JAS+HOG^2 [65]	94.8%
DL-GSGC+TPM [60]	96.7%
Proposed MMMP	**98.2%**

Table 3.6 Performance Comparison for MSR Action 3D Dataset Over All Action Classes

Method (protocol of [110])	Accuracy
Depth HOG [126] (as reported in [112])	85.5%
Actionlet Ensemble [110]	88.2%
HON4D [67]	88.9%
DSTIP [120]	89.3%
Lie Group [100]	89.5%
HOPC [73]	91.6%
Max Margin Time Warping [112]	92.7%
Proposed MMMP	**93.1%**

3.3.4 3D Action Pairs Dataset

Here we apply our full multimodal multipart learning method using all three available modalities of features. As shown in Table 3.7, the proposed method, outperforms all others and saturates the benchmark by achieving the perfect performance level on this dataset.

Table 3.7 Performance Comparison for 3D Action Pairs Dataset

Method	**Accuracy**
Depth HOG [126] (as reported in [112])	66.11%
Actionlet Ensemble [110] (as reported in [112])	82.22%
HON4D [67]	96.67%
Max Margin Time Warping [112]	97.22%
HOG3D-LLC [70]	98.33%
HOPC [73]	98.33%
Proposed MMMP	**100.0%**

Table 3.8 Performance Comparison for input features and proposed method on Cross-Subject evaluation protocol of NTU RGB+D Action Dataset. The dimensionality of the input features are reduced by PCA.

Method/Feature	**Accuracy**
FTP on Skeletons	58.62%
FTP on LOP	36.67%
LLC on HON4D	53.33%
Proposed MMMP	**63.97%**

3.3.5 NTU RGB+D Action dataset

NTU RGB+D Action is a large scale benchmark dataset for 3D action recognition. The details about this dataset are provided in section 5.2. The large scale of this dataset makes it possible to study the proposed algorithm further and verify other aspects of it in more detail.

For our experiments in this section, we use the same set of features used for the other datasets, which are explained in section 3.3.1. Our studies are done on the cross-subject evaluation protocol of this dataset. To better study the behavior of the method regarding body joints, we removed the features of the auxiliary part introduced in the normalization step. Since the number of training samples of this dataset are more than others in orders of magnitude, we apply PCA to reduce the dimensionality of each input modality and keep the entire inputs matrix in a feasible size.

Table 3.8 shows the performance of the proposed MMMP method in contrast with the input features used.

To ease the study of next experimental results, we mention the list of action classes and joint names here. The 60 action classes in NTU RGB+D dataset are: 1-drinking, 2-eating, 3-brushing teeth, 4-brushing hair, 5-dropping, 6-picking up, 7-throwing, 8-sitting down, 9-standing up (from sitting position), 10-clapping, 11-reading, 12-writing, 13-tearing up paper, 14-wearing jacket, 15-taking off jacket, 16-wearing a shoe, 17-taking off a shoe, 18-wearing on glasses, 19-taking off glasses, 20-puting on a hat/cap, 21-taking off a hat/cap, 22-cheering up, 23-hand waving, 24-kicking something, 25-reaching into self pocket, 26-hopping, 27-jumping up, 28-making/answering a phone call, 29-playing with phone, 30-typing, 31-pointing to something, 32-taking selfie, 33-checking time (on watch), 34-rubbing two hands together, 35-bowing, 36-shaking head, 37-wiping face, 38-saluting, 39-putting palms together, 40-crossing hands in front. 41-sneezing/coughing, 42-staggering, 43-falling down, 44-touching head (headache), 45-touching chest (stomachache/heart pain), 46-touching back (back-pain), 47-touching neck (neck-ache), 48-vomiting, 49-fanning self. 50-punching/slapping other person, 51-kicking other person, 52-pushing other person, 53-patting other's back, 54-pointing to the other person, 55-hugging, 56-giving something to other person, 57-touching other person's pocket, 58-handshaking, 59-walking towards each other, and 60-walking apart from each other.

The 25 body joints in this dataset are illustrated in Fig. 5.2. The names of the joints are: 1-base of the spine, 2-middle of the spine, 3-neck, 4-head, 5-left shoulder, 6-left elbow, 7-left wrist, 8-left hand, 9-right shoulder, 10-right elbow, 11-right wrist, 12-right hand, 13-left hip, 14-left knee, 15-left ankle, 16-left foot, 17-right hip, 18-right knee, 19-right ankle, 20-right foot, 21-spine, 22-tip of the left hand, 23-left thumb, 24-tip of the right hand, and 25-right thumb.

Fig. 3.2 illustrates the ℓ_2 norm of the group of the weights corresponding to each joint of the body for all the action classes. This shows how each body joint participates in the weights learned for each action class. As expected, for each action, a small number of joints have notable weights which can be considered activated. Some of the joints are activated for most of the actions. They are 8-left hand, 12-right hand, 22-tip of the left hand, and 24-tip of the right hand. This is reasonable, because the articulated motion of the hands are highly discriminative for most of the human actions.

In addition, for each action class, we can observe the activated joints. For example, actions 24-kicking something, 26-hopping, 27-jumping up, and 43-falling down have the block of lower-body joints (number 13 to 20) activated. Unlike most of the other actions, 35-bowing does not activate hand joints, instead it has 1-base of the spine, 13-left hip, and 17-right hip as important joints.

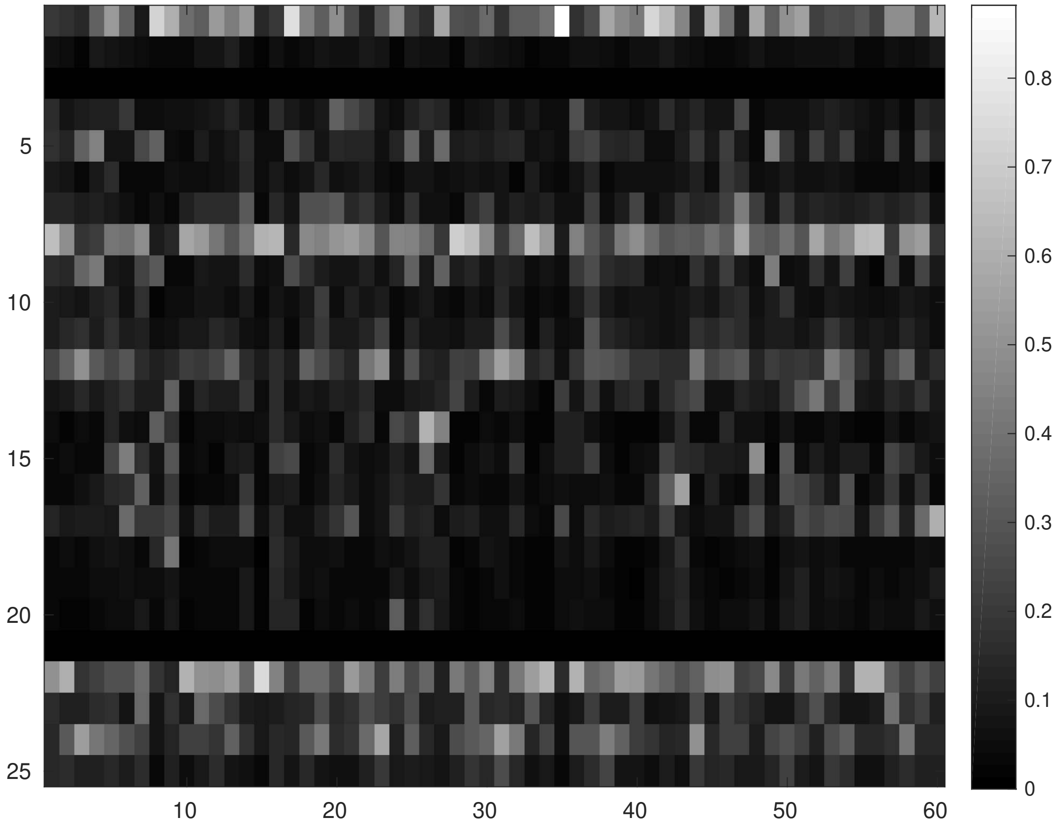


Fig. 3.2 ℓ_2 norm values of the weights assigned to each body part, for all of the action classes. Horizontal axis shows the 60 action classes. Vertical axis denotes the index of joints.

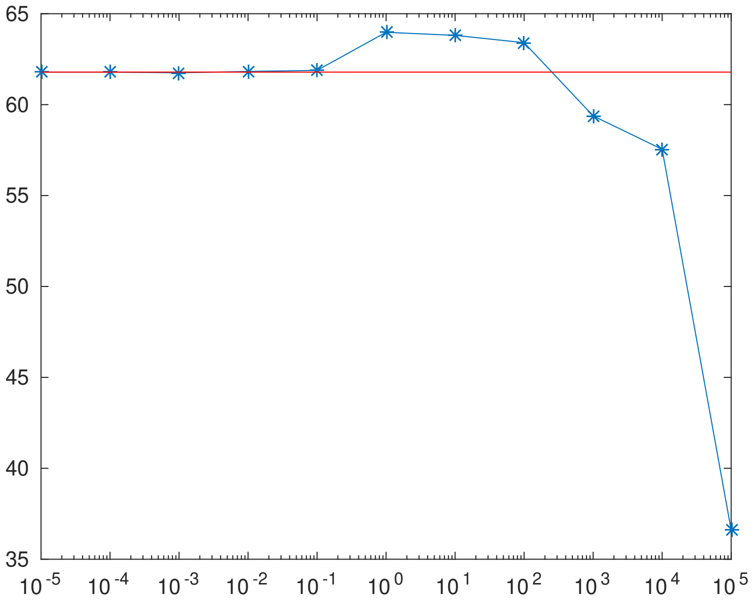


Fig. 3.3 Comparison of the MMMP performance for different values of parameter $\widehat{\lambda_1}$ on the cross-subject evaluation protocol of NTU RGB+D Action Dataset. The red line shows the valued for $\widehat{\lambda_1} = 0$. The vertical axis shows the performance in percents and the horizontal axis shows the values for the evaluated parameter.

To further study the sensitiveness of the proposed method on the values of different parameters, we evaluate the final performance of the MMMP method on a range of different values for each of the hyper-parameters: $\widehat{\lambda_1}, \widehat{\lambda_2}, \lambda_1, \lambda_2$, and λ_3. Figures 3.3-3.7 illustrate the results of these evaluations, which show the robustness of the proposed MMMP method on a large range of values for each parameter, although very large values for each of the parameters can deviate the optimization from the desired optimum point. Comparing the values with the red lines which denote the performance of the method with the zero-valued parameter, show all of the corresponding terms are effective in the overall performance of the method. Specifically, Fig. 3.7 illustrates the effect of the values of λ_3 in comparison to its zero value. This shows the importance of the last term in equation 3.14 which keeps the learning weights close to the partially optimum values learned by multipart method separately for each modality.

Finally, Table 3.9 shows the performance of the proposed MMMP method in contrast to other depth-based action recognition methods.

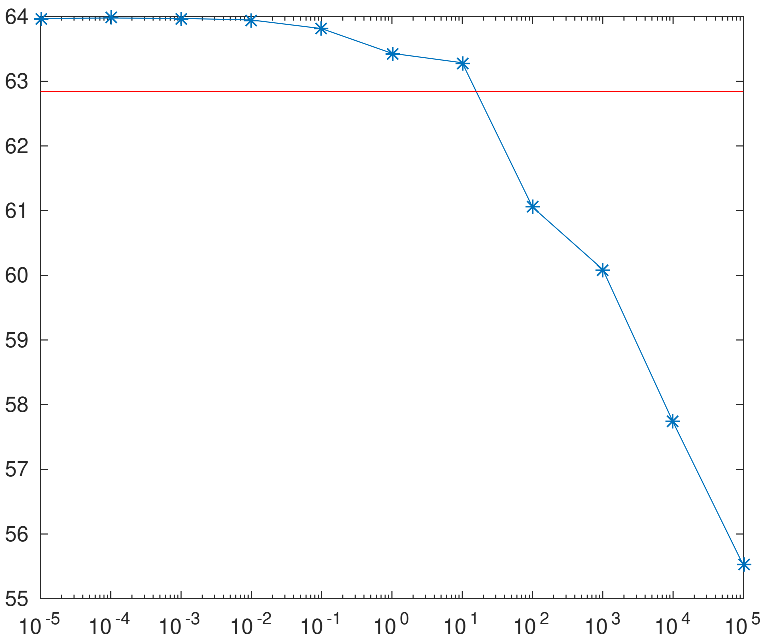


Fig. 3.4 Comparison of the MMMP performance for different values of parameter $\widehat{\lambda_2}$ on the cross-subject evaluation protocol of NTU RGB+D Action Dataset. The red line shows the valued for $\widehat{\lambda_2} = 0$. The vertical axis shows the performance in percents and the horizontal axis shows the values for the evaluated parameter.

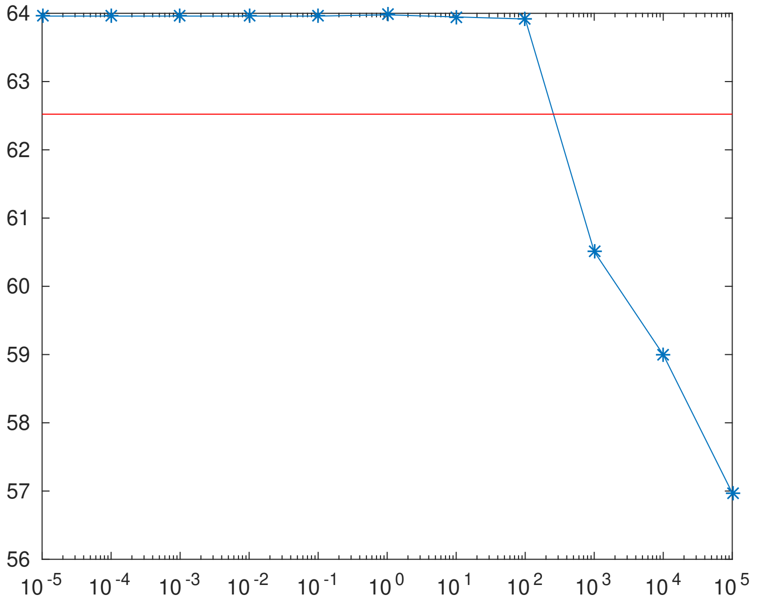


Fig. 3.5 Comparison of the MMMP performance for different values of parameter λ_1 on the cross-subject evaluation protocol of NTU RGB+D Action Dataset. The red line shows the valued for $\lambda_1 = 0$. The vertical axis shows the performance in percents and the horizontal axis shows the values for the evaluated parameter.

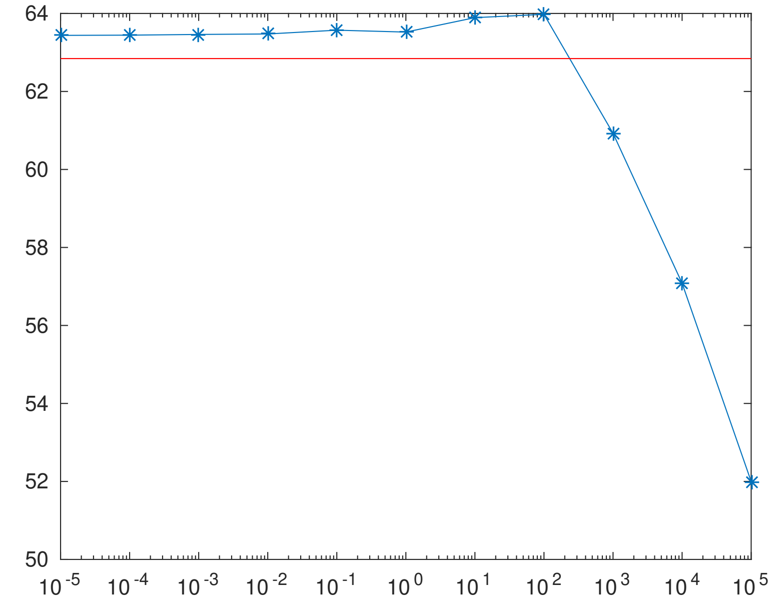


Fig. 3.6 Comparison of the MMMP performance for different values of parameter λ_2 on the cross-subject evaluation protocol of NTU RGB+D Action Dataset. The red line shows the valued for $\lambda_2 = 0$. The vertical axis shows the performance in percents and the horizontal axis shows the values for the evaluated parameter.

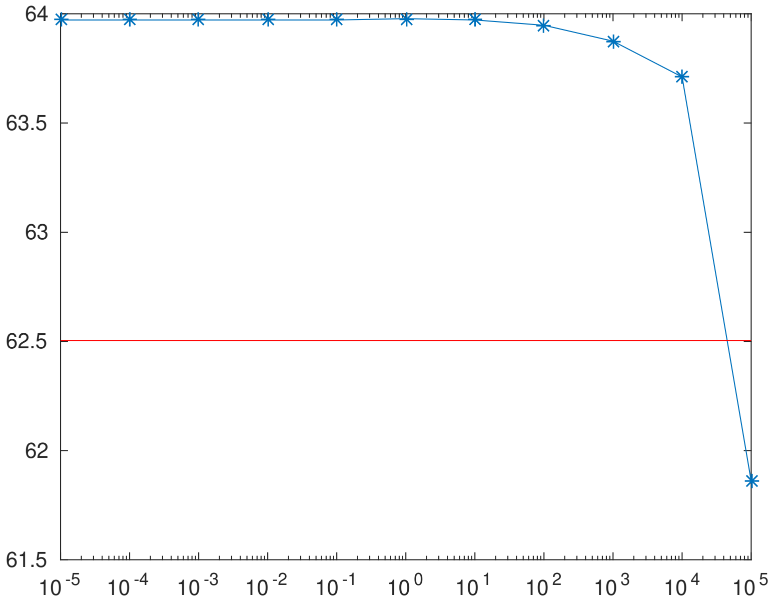


Fig. 3.7 Comparison of the MMMP performance for different values of parameter λ_3 on the cross-subject evaluation protocol of NTU RGB+D Action Dataset. The red line shows the valued for $\lambda_3 = 0$. The vertical axis shows the performance in percents and the horizontal axis shows the values for the evaluated parameter.

Method	Accuracy
HOG^2 [65]	32.24%
Super Normal Vector [125]	24.56%
HON4D [67]	30.56%
Lie Group [100]	50.08%
Skeletal Quads [24]	38.62%
FTP Dynamic Skeletons [34]	60.23%
Proposed MMMP	**63.97%**

Table 3.9 Comparison of the proposed MMMP with other methods on the cross-subject evaluation protocol of NTU RGB+D Action Dataset.

3.4 Chapter Summary

This chapter presents a new multimodal multipart learning approach for action classification in depth sequences. We show that a sparse combination of multimodal part-based features can effectively and discriminatively represent all the available action classes at hand. Based on the nature of the problem, we utilize a heterogeneous set of features from skeleton based 3D joint trajectories, depth occupancy patterns and histograms of depth surface normals and show the proper way of using them as multimodal features set for each part.

The proposed method does the group feature selection, weight regularization, and classifier learning in a consistent optimization step. It applies the proposed hierarchical mixed norm to model the proper structure of multimodal multipart input features by applying a diversity norm over the coefficients of each part-modality group, linking different modalities of each part by a magnitude based norm, and utilizing a soft part selection by a sparsity inducing norm.

The provided experimental evaluations on four challenging depth based action recognition datasets show the proposed method can successfully apply the structure of the input features into a concurrent group feature selection and learning and confirm the strengths of the suggested framework compared to other methods.

A part of the work presented in this chapter is published in [83].

Chapter 4

Deep Correlation-Independence Analysis for RGB+D Action Recognition

In previous chapter, we studied the fusion of different depth-based features for action classification. However, in realistic settings of Kinect-based human activity analysis, RGB videos are also available in addition to depth-maps.

Since the analysis of activities on traditional RGB videos has been extensively studied during last decade, it is reasonable to move towards RGB+D multimodality analysis in order to achieve better understanding and higher performance accuracy in human action recognition problem.

Therefore, in this chapter we focus on multimodal RGB+D based action recognition and propose a new action analysis framework which analyses the input RGB+D features based on their mutually correlated and their modality-specific components.

4.1 Introduction

Recent advances in hand-crafted [107, 68] and convnet-based [86] feature extraction and analysis of RGB action videos achieved impressive performance. They generally recognize action classes based on appearance and motion patterns in videos. The major limitation of RGB sequences is the absence of 3D structure from the scene. Although some works are done towards this direction [23], recovering depth from RGB in general is an underdetermined problem. As a result, depth sequences provide an exclusive modality of information about the 3D structure of the scene, which suits the problem of activity analysis [67, 73, 100, 110, 112, 120, 128]. This complements the textural and appearance information from RGB. Our purpose in this chapter is to analyze the multimodal RGB+D signals for identifying the strengths of respective modalities through teasing out their mutual and independent components and to utilize them for improving the classification of human actions.

Having multiple sources of information, one can find a new space of common components which can be more robust than any of the input features. Through linear projections, Canonical Correlation Analysis (CCA) [30, 33] gives us the correlated form of input modalities which in essence is a robust representation of multimodal signals. However, the downside of CCA is the linearity limitation. Kernel Canonical Correlation Analysis (KCCA) [44] extended this idea into nonlinear kernel-based projections, which is still limited to the representation capacity of the kernel's space and is not able to disentangle the high-level nonlinear complexities between the input modalities. Further, the traditional solutions of CCA and KCCA are to solve the maximization of correlation between the projected vectors analytically, which does not scale well with the size of the data.

To overcome these limitations, a new deep autoencoder-based nonlinear correlation analysis network is proposed to discover the common components of input RGB+D signals.

Besides the correlated components, each input modality has independent features which carry discriminative information for the recognition task. In this respect, we can enhance the representation by incorporating the independent components of respective modalities [13, 79]. Based on this intuition, at each layer our deep network factorizes the multimodal input features into their correlated and mutually independence components. By stacking such layers, we further decode the complex and highly nonlinear representations of the input modalities in a nonlinear fashion.

Across the layers, our deep correlation-independence analysis extracts a set of structured features which consist of hierarchically factorized correlated and independent components. The correlated components are robust against noise and missing information between the modalities, and the independent components carry the modality-specific features. To effectively perform recognition tasks on our structured features, we design a structured sparsity-based learning framework. With different mixed norms, features of each component can be grouped together and group selection can be applied to learn a better classifier. We also show that the advantage of our learning framework is more significant as the network gets deeper.

One of the most related methods to this work is [35] which factorized the input features to shared and private components by applying structured sparsity, for the task of multi-view learning on human pose estimation, with linearity assumption. In addition, Cai *et al.* [13] proposed a nonlinear factorization of the features into common and individual components, towards a better representation of features for action recognition. They utilized mixture models to add nonlinearity to linear probabilistic CCA [5]. In contrast, the proposed analysis technique in this chapter stacks layers of nonlinear correlation analysis to progressively disentangle highly nonlinear correlations between the input features.

While learning frameworks in [103–106] applied structured sparsity for other similar tasks, our structured sparsity learning machine extends the sparse selection into two levels of concurrent component and layer selection, which is more suited to the hierarchical outputs from our deep factorization network.

There are other works which utilized deep networks for multimodal learning. Srivastave and Salakhutdinov [89] and Ngiam *et al.* [62] used deep Boltzmann machines (DBM) for finding a common space representation for two input modalities, and predict one modality from the other. Andrew *et al.* [4] proposed a deep canonical correlation analysis (DCCA) network with two stacks of deep embedding followed by a canonical correlation analysis(CCA) on top layer.

The proposed method in this chapter is different from these works in two major aspects. First, the previous work performed the correlation analysis in just one layer of the deep network, but our proposed method performs correlation analysis in every single layer. Second, we incorporate independent components in each layer to maintain all the modality specific information, at each layer. Our solution is general and can be applied on any type of RGB and depth based features to analyze their correlated and independent components.

The rest of this chapter is organized as follows. Section 4.2 describes the proposed deep component factorization network. Section 4.3 presents our structured sparsity-based classification framework for factorized components. Section 4.4 introduces a CCA-based baseline method for multimodal RGB+D analysis and factorization. Section 4.5 provides our experimental results, and section 4.6 concludes the chapter.

4.2 Deep Correlation-Independence Analysis

We have two sets of features extracted from different modalities of data (RGB and depth signals) as our input for the task of action classification. State-of-the-art RGB based features [102, 107] include 2D motion patterns and appearance information of objects and scenes. On the other hand, various depth-based features [67, 73, 110, 120] encode 3D shape and motion information, without appearance and texture details.

Consequently, it is beneficial to fuse the complementary RGB and depth-based features for better performance in action analysis.

There are different techniques for feature fusion. The choice of fusion strategy should rely on dependency of features [79]. Since RGB and depth based features encode an entangled combination of common and modality-specific information of the observed action, they are neither independent nor fully correlated. Therefore, it is reasonable to embed the input data into a space of factorized correlated components and independent components. The combination of the correlated and independent components in input features can be very complex and highly nonlinear. To disentangle them, we stack layers of nonlinear autoencoder-based component factorization to form a deep correlation-independence analysis network.

In this section, we first introduce our basic framework of correlation-independence analysis for multimodal signal factorization, then describe the deep network of stacked layers, where each layer performs correlation-independence analysis and collectively produce a hierarchical set of correlated and independent components.

4.2.1 Single Layer Correlation-Independence Analysis

Let us notate input RGB features by $\mathbf{X}_r$ and depth features by $\mathbf{X}_d$. We propose to factorize each input feature pattern into two spaces: first, common component space which corresponds to the highest correlation with the other modality $(\mathbf{Y}_r, \mathbf{Y}_d)$, and second, its independent feature component space $(\mathbf{Z}_r, \mathbf{Z}_d)$:

$$\begin{bmatrix} \mathbf{Y}_r \\ \mathbf{Y}_d \\ \mathbf{Z}_r \\ \mathbf{Z}_d \end{bmatrix} = g(\mathbf{X}_r, \mathbf{X}_d; \Omega) \tag{4.1}$$

where Ω is the set of model parameters that will be learned from the training data. We propose a sparse autoencoder-based network as the $g(.)$ function, as illustrated in Fig. 4.1.

Feature vectors of each modality are factorized into $\mathbf{Y}$ and $\mathbf{Z}$ which represent shared and individual components of each modality respectively. Each component is derived from a linear projection of the input features followed by a nonlinear activation.

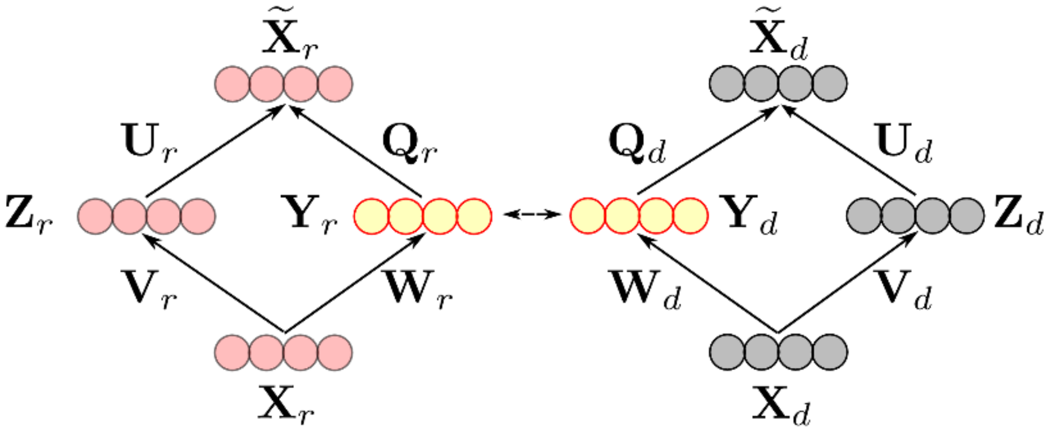


Fig. 4.1 Illustration of the proposed single layer correlation-independence analysis. $\mathbf{X}_r$ and $\mathbf{X}_d$ are input RGB and depth based features. We factorize each input feature into correlated $(\mathbf{Y})$ and independent $(\mathbf{Z})$ components by enforcing the $\mathbf{Y}$ components to be close, and the input features to be reconstructible from derived components.

Mathematically:

$$\mathbf{Y}_r = f(\mathbf{W}_r\mathbf{X}_r + \mathbf{b}_{Y_r}\mathbf{1}^n) \tag{4.2}$$

$$\mathbf{Z}_r = f(\mathbf{V}_r\mathbf{X}_r + \mathbf{b}_{Z_r}\mathbf{1}^n) \tag{4.3}$$

in which $f(.)$ is a nonlinear activation function. We used sigmoid scaling in our implementation. $\mathbf{W}_.$ and $\mathbf{V}_.$ are learning weights, and $\mathbf{b}_.$ are learning bias terms.

Similarly, for the depth based input, we have:

$$\mathbf{Y}_d = f(\mathbf{W}_d\mathbf{X}_d + \mathbf{b}_{Y_d}\mathbf{1}^n) \tag{4.4}$$

$$\mathbf{Z}_d = f(\mathbf{V}_d\mathbf{X}_d + \mathbf{b}_{Z_d}\mathbf{1}^n) \tag{4.5}$$

To prevent output degeneration, we expect the original features to be reconstructible from their factorized components [48]:

$$\begin{aligned}\widetilde{\mathbf{X}}_r &= f(\begin{bmatrix}\mathbf{Q}_r \ \mathbf{U}_r\end{bmatrix}\begin{bmatrix}\mathbf{Y}_r \\ \mathbf{Z}_r\end{bmatrix} + \mathbf{b}_{\widetilde{X}_r}\mathbf{1}^n) \\ &= f(\mathbf{Q}_r\mathbf{Y}_r + \mathbf{U}_r\mathbf{Z}_r + \mathbf{b}_{\widetilde{X}_r}\mathbf{1}^n)\end{aligned} \tag{4.6}$$

$$\begin{aligned}\widetilde{\mathbf{X}}_d &= f(\begin{bmatrix}\mathbf{Q}_d \ \mathbf{U}_d\end{bmatrix}\begin{bmatrix}\mathbf{Y}_d \\ \mathbf{Z}_d\end{bmatrix} + \mathbf{b}_{\widetilde{X}_d}\mathbf{1}^n) \\ &= f(\mathbf{Q}_d\mathbf{Y}_d + \mathbf{U}_d\mathbf{Z}_d + \mathbf{b}_{\widetilde{X}_d}\mathbf{1}^n)\end{aligned} \tag{4.7}$$

where $\mathbf{Q}_.$ and $\mathbf{U}_.$ are learning weights, and $\mathbf{b}_.$ are learning bias terms. Now we can formulate the desired component factorization into an optimization problem with the cost function:

$$\begin{aligned}\Omega^* = \underset{\Omega}{argmin}\ & \Delta(\mathbf{Y}_r, \mathbf{Y}_d) + \lambda\ \|\Omega\|_2 \\ & + \zeta_r\ \Delta(\mathbf{X}_r, \widetilde{\mathbf{X}}_r) + \zeta_d\ \Delta(\mathbf{X}_d, \widetilde{\mathbf{X}}_d) \\ & + \alpha_r\ \Psi(\mathbf{Y}_r; \rho_Y) + \alpha_d\ \Psi(\mathbf{Y}_d; \rho_Y) \\ & + \beta_r\ \Psi(\mathbf{Z}_r; \rho_Z) + \beta_d\ \Psi(\mathbf{Z}_d; \rho_Z) \end{aligned} \tag{4.8}$$

where $\Omega = \{\mathbf{W}_., \mathbf{V}_., \mathbf{Q}_., \mathbf{b}_.\}$ is the set of all parameters, and $[\lambda, \zeta_., \alpha_., \beta_.]$ are hyper-parameters of trade-off between terms.

The first term in (4.8) enforces the correlated components of the two modalities ($\mathbf{Y}_r$ and $\mathbf{Y}_d$) to be as close as possible. We formulate this term as the Frobenius norm of the difference between two matrices:

$$\Delta(\mathbf{Y}_r, \mathbf{Y}_d) = \|\mathbf{Y}_r - \mathbf{Y}_d\|_F \tag{4.9}$$

The second term is the general weight regularization term, applied on the projection weights to prevent networks from overfitting training data.

The reconstruction costs are represented as $\Delta(\mathbf{X}_r, \widetilde{\mathbf{X}}_r)$ and $\Delta(\mathbf{X}_d, \widetilde{\mathbf{X}}_d)$ to prevent the model from degeneration. Here, we use Frobenius norm (the same as (4.9)) of the reconstruction error for the reconstruction cost term.

Similar to the traditional sparse auto-encoder frameworks, applying sparsity on the learned hidden units is crucial for learning and discovery of hidden structures of the input data, especially when the size of the hidden representations are not much less than the size of the input vectors.

Last four terms of (4.8) are sparsity penalty terms over $\mathbf{Y}$ and $\mathbf{Z}$ hidden representations separately. It has been shown in [49, 77] that applying sparsity on the features of $\mathbf{Y}$ and $\mathbf{Z}$ will help to improve the learning capability, especially when components are overcomplete. As our sparsity penalty functions (Ψ), we use KL-divergence term, applied between $\mathbf{Y}$ components the sparsity parameters ρ_Y, and between $\mathbf{Z}$ components and ρ_Z .

It is worth pointing out, since the proposed framework is built on a sparse autoencoder-like scheme and has sigmoid scaling nonlinearity, it is necessary to apply PCA whitening on the input matrices $\mathbf{X}_r$ and $\mathbf{X}_d$ and scale their elements into the range of $[0, 1]$.

In this formulation, the independence of $\mathbf{Z_d}$ and $\mathbf{Z_r}$ components is enforced implicitly. The similarity inducing norm pushes the correlated components of the two modalities to move inside $\mathbf{Y}$ components. Therefore, we expect the remaining features in each of $\mathbf{Z}$ components to be highly different across the modalities.

4.2.2 Deep Correlation-Independence Analysis

State-of-the-art RGB and depth based features for action recognition, are extracted by multiple linear and nonlinear layers of projection, embedding, spatial and temporal pooling, statistical distribution encodings, *e.g.* bag of visual words [87], Fisher Vectors [69] or Fourier temporal pyramids in [110]. Hence the correlated components between modalities can lie on highly complex and nonlinear subspaces of input data, and one layer of the proposed correlation-independence analysis cannot decode these correlated components well.

By cascading multiple correlation-independence analysis layers, we build a deep network to further factorize input features based on their higher orders of shared and independent information between modalities. To do so, we feed $\mathbf{Y}$ components of the previous layer as multimodal inputs of the current layer and apply the same method with new learning parameters in order to further factorize the features. As illustrated in Fig. 4.2, each layer extracts independent components of the modalities and passes the shared components for further factorization in the next layer:

$$\begin{bmatrix} \mathbf{Y}_r^{(i)} \\ \mathbf{Y}_d^{(i)} \\ \mathbf{Z}_r^{(i)} \\ \mathbf{Z}_d^{(i)} \end{bmatrix} = \begin{cases} g(\mathbf{X}_r, \mathbf{X}_d; \Omega^{(i)}) & \text{if } i = 1 \\ g(\mathbf{Y}_r^{(i-1)}, \mathbf{Y}_d^{(i-1)}; \Omega^{(i)}) & \text{if } i > 1 \end{cases} \tag{4.10}$$

For almost all of the depth-based action recognition datasets, the number of training samples are not high enough compared to the number of learning weights of the overall network. In this way, optimizing the entire network in an end-to-end manner leads to overfitting. Therefore, our deep network is trained greedily and layer-wise, similar to [6, 31, 50]. In other words, the optimization of each layer is started upon the convergence of the previous layer's training.

Upon training of the deep network, each input sample will be factorized into a pair of independent components $(\mathbf{Z}_r^{(i)}, \mathbf{Z}_d^{(i)})$ for each layer $i \in [1, .., l]$, plus the concatenation of last layer's correlated components $(\mathbf{Y}_r^{(l)}, \mathbf{Y}_d^{(l)})$.

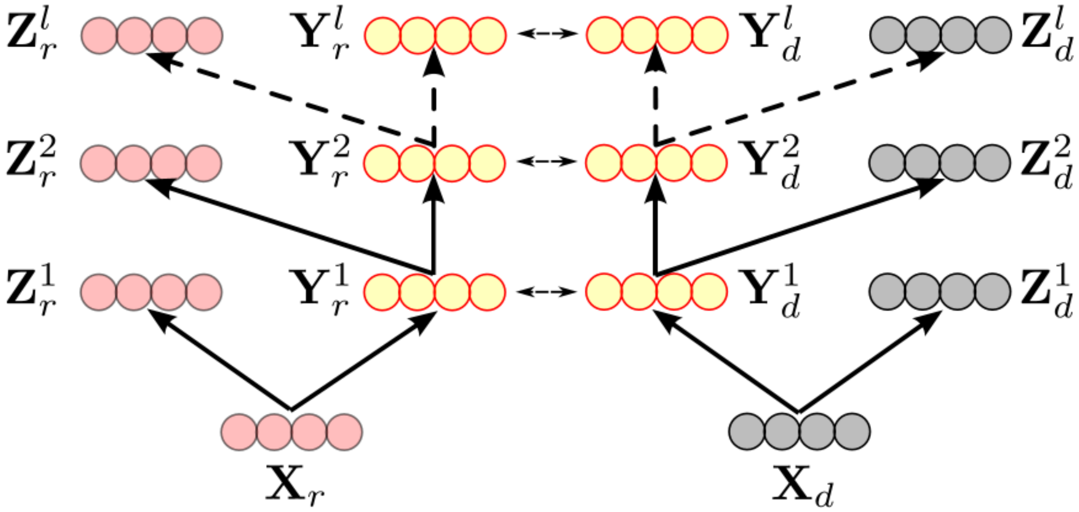


Fig. 4.2 Cascading factorization layers to a deep correlation-independence network. To disentangle the highly nonlinear combination of correlated-independent components, factorization layers are stacked by feeding the **Y** components of each layer as inputs of the next layer.

4.2.3 Convolutional Correlation-Independence Analysis

By limiting our analysis into holistic RGB+D features, we may lose discriminative local information in both modalities. In addition, local features also have dependencies across modalities and their correlation-independence analysis (CIA) is important. Therefore, as depicted in Fig. 4.3, we first train the local deep CIA network (CIA^L) on local RGB+D features of smaller cubes of training samples videos. Then we apply the learned CIA^L to decompose the features of all local cubes. By concatenating all the factorized components of local cubes together with corresponding holistic features of video samples, we build the input for the holistic CIA network (CIA^H), similar to [47]. The inputs of CIA^H are PCA whitened and scaled into the range of $[0,1]$.

Overall, we have $L = l_1 + l_2$ layers of factorization where l_1 and l_2 are the number of layers in CIA^L and CIA^H networks respectively. By applying the trained local-holistic networks into the features of each video sample, we have a set of $2L+1$ independent components:

$$\mathbf{A} = \{(\mathbf{Z}_r^1)^T, (\mathbf{Z}_d^1)^T, ..., (\mathbf{Z}_r^L)^T, (\mathbf{Z}_d^L)^T, (\mathbf{Y}^L)^T\}^T \qquad (4.11)$$

where $\mathbf{Y}^L = \begin{bmatrix} \mathbf{Y}_r^L \\ \mathbf{Y}_d^L \end{bmatrix}$ is the concatenation of last layer's correlated components.

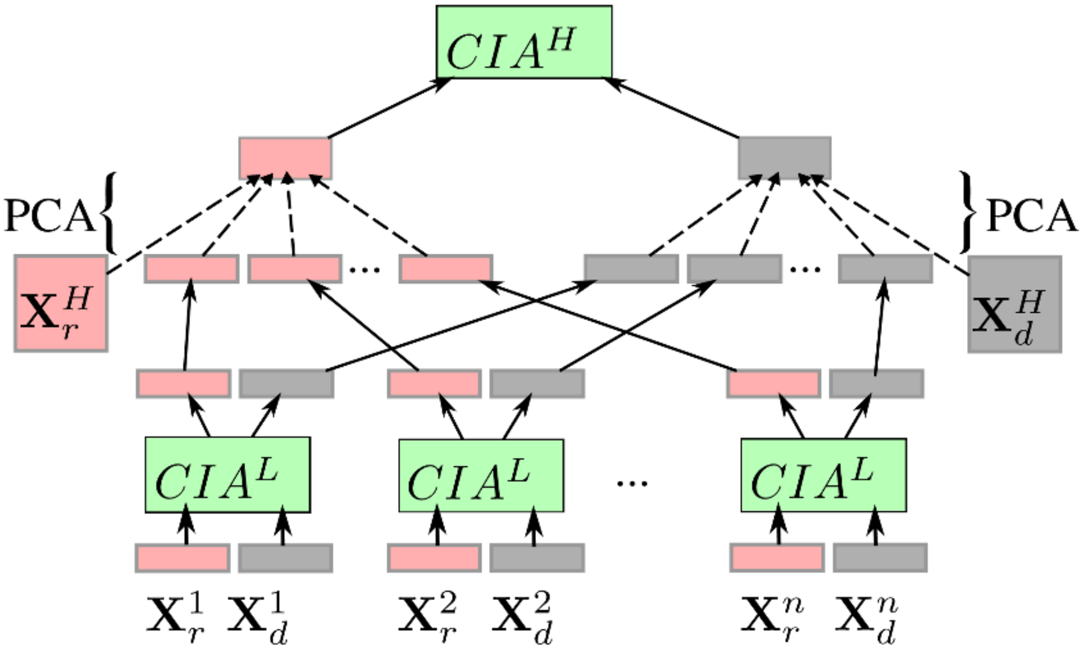

CIA^H
PCA
PCA
$\mathbf{X}_r^H$
$\mathbf{X}_d^H$
CIA^L
CIA^L
CIA^L
$\mathbf{X}_r^1$ $\mathbf{X}_d^1$
$\mathbf{X}_r^2$ $\mathbf{X}_d^2$
$\mathbf{X}_r^n$ $\mathbf{X}_d^n$

Fig. 4.3 Schema of our convolutional and holistic networks of correlation-independence analysis (*CIA*). We divide each video into n local cubes. Local features $\mathbf{X}_r^i$ and $\mathbf{X}_d^i$ are extracted from the i^{th} cube. Convolutional network (denoted as CIA^L) is trained and then applied to decompose local features. The factorized components are then combined with holistic features $\mathbf{X}_r^H$ and $\mathbf{X}_d^H$. This combination undergoes PCA and is fed into the holistic network (denoted as CIA^H) as its multimodal input.

4.2.4 Optimization Algorithm

The proposed formulation of cost function (4.8) is not a convex function of training parameters. Therefore, optimization of the learning parameters is not feasible in a single step. We iteratively optimize subsets of the parameters while keeping others fixed to achieve a suboptimal solution which is already shown effective in different applications [122].

Specifically, the learning parameters of each layer can be divided into two subsets. First are the ones defined for projection and reconstruction of the correlated components $\mathbf{Y}_{.}$, and second consists of similar parameters for individual component $\mathbf{Z}_{.}$. These two sets are:

$$\Omega_Y = \{\mathbf{W}_r, \mathbf{W}_d, \mathbf{Q}_r, \mathbf{Q}_d, \mathbf{b}_{Y_r}, \mathbf{b}_{Y_d}, \mathbf{b}_{\widetilde{X}_r}, \mathbf{b}_{\widetilde{X}_d}\} \tag{4.12}$$

$$\Omega_Z = \{\mathbf{V}_r, \mathbf{V}_d, \mathbf{U}_r, \mathbf{U}_d, \mathbf{b}_{Z_r}, \mathbf{b}_{Z_d}, \mathbf{b}_{\widetilde{X}_r}, \mathbf{b}_{\widetilde{X}_d}\} \tag{4.13}$$

Now, to optimize the overall cost, we first fix Ω_Z (except $\mathbf{b}_{\widetilde{X}_.}$) and minimize the cost function (4.8) regarding Ω_Y. Then fix parameters of Ω_Y (except $\mathbf{b}_{\widetilde{X}_.}$) and optimize regarding Ω_Z and repeat this iteratively to converge into a suboptimal point.

In our implementation, all the optimization steps are done by “L-BFGS” algorithm using off-the-shelf “minFunc” software [80].

4.3 Structured Sparsity Learning Machine

Previous correlation-independence analysis steps were all unsupervised and applied just based on the dependency of the two modalities. As a result, the factorized features of each component are not guaranteed to be equally discriminative for the following classification step. Hence we adopt the structured sparsity regularization method of [104, 106] aiming to select a number of components/layers sparsely to achieve more robust classification. Since the features of each component are highly correlated, our structured sparsity regularizer bounds the weights of the features inside each component to become activated or deactivated together.

Mathematically, we want to learn a linear projection matrix $\mathbf{B}$ to project our hierarchically factorized features $\mathbf{A}$ (see align 4.11), to a class assignment matrix $\mathbf{F}$

defined as:

$$f_i^j = \begin{cases} 1 & \text{if } j^{th} \text{ sample belongs to the } i^{th} \text{ class} \\ 0 & \text{otherwise} \end{cases} \tag{4.14}$$

so that $\mathbf{A}^T\mathbf{B}$ would be as close as possible to $\mathbf{F}$.

Each column of $\mathbf{A}$ consists of $2L+1$ components of features for each training sample. We use the notation $\mathbf{A}^G$ to denote the rows of $\mathbf{A}$ which include the features of component G. Variable G is the index of the group of the feature elements and can take values between 1 and $2L+1$ or their corresponding component labels. Correspondingly, columns of $\mathbf{B}$ have the same structure, and we denote the G^{th} component's parameters as $\mathbf{B}^G$. We refer to the i^{th} column of $\mathbf{B}$ as $\mathbf{b}_i$ which is the projection to our binary classifier for the i^{th} action. Finally $\mathbf{b}_i^G$ refers to the i^{th} column of $\mathbf{B}^G$.

Our classifier is formulated as another optimization problem with the cost function below.

$$\mathbf{B}^* = \underset{\mathbf{B}}{argmin} \left\|\mathbf{A}^T\mathbf{B} - \mathbf{F}\right\|_F^2 + \gamma_E \ ||\mathbf{B}||_{G_E} + \gamma_L \ ||\mathbf{B}||_{G_L} + \gamma_W \ ||\mathbf{B}||_F \tag{4.15}$$

The first term of (4.15) is the squared distance norm between the projections and the ground truth classification labels of $\mathbf{F}$. The second term, is the mixed norm to group and regularize the weights of each component together and apply sparsity between them to enforce a component selection regime. Similarly, the third term applies a layer-wise selection and regularization within each layer. The last norm in (4.15) is a general weight decay regularizer to prevent the entire classifier from overfitting.

Component-wise regularizer norm, $||\mathbf{B}||_{G_E}$, groups the weights of each component by applying a ℓ_2 norm. Then applies the component selection by a ℓ_1 norm over the ℓ_2 values of all components. Mathematically:

$$\begin{aligned} ||\mathbf{B}||_{G_E} &= \sum_{i=1}^{c} \sum_{G=1}^{2L+1} \left\|\mathbf{b}_i^G\right\|_2 \\ &= \sum_{i=1}^{c} \sum_{j=1}^{L} \left(\left\|\mathbf{b}_i^{Z_r^j}\right\|_2 + \left\|\mathbf{b}_i^{Z_d^j}\right\|_2 \right) + \sum_{i=1}^{c} \left\|\mathbf{b}_i^{Y^L}\right\|_2 \end{aligned} \tag{4.16}$$

where c is the number of class labels.

This mixed norm dictates the component-wise weight learning regarding their discriminative strength for each action class. Since it applies ℓ_2 norm inside the components and ℓ_1 norm between them, it regularizes the weights within each component, while sparsely selects discriminative components for different classes.

On the other hand, a layer-wise group selection can also be beneficial, because discriminative features may become factorized in some layers of our hierarchical deep network. Based on this intuition, we apply another group sparsity mixed norm to enforce layer selection. Similar to G_E norm, our layer selection norm (G_L) groups the learning parameters corresponding to the components of each layer of the network, and applies ℓ_1 sparsity between them:

$$\|\mathbf{B}\|_{G_L} = \sum_{i=1}^{c} \sum_{j=1}^{L} \left\| \begin{bmatrix} \mathbf{b}_i^{Z_r^j} \\ \mathbf{b}_i^{Z_d^j} \end{bmatrix} \right\|_2 + \sum_{i=1}^{c} \left\| \mathbf{b}_i^{Y^L} \right\|_2 \tag{4.17}$$

Similar to previous section, this optimization is also done using “L-BFGS” algorithm. Upon training the classifier and finding the optimal $\mathbf{B}^*$, we classify each testing sample with exemplar features $\mathbf{a_q}$ as:

$$h(\mathbf{a}_q) = \underset{c}{argmax} \ \langle \ \mathbf{a}_q, \mathbf{b}_c^* \ \rangle \tag{4.18}$$

4.4 CCA-RICA Factorization as a Baseline Method

As a baseline to the proposed method to perform the correlation-independence analysis of the RGB+D inputs, we combined canonical correlation analysis (CCA) [33, 30] and reconstruction independent component analysis (RICA) [48], to extract correlated and independent components of input features. In this section we describe this baseline method.

We use the notation $\mathbf{X}_r$ to represent input local RGB features, and $\mathbf{X}_d$ for corresponding local depth features. We define the linear projections of the two input features as:

$$\mathbf{Y}_r = \mathbf{W}_{r,c}\mathbf{X}_r \qquad , \qquad \mathbf{Y}_d = \mathbf{W}_{d,c}\mathbf{X}_d \tag{4.19}$$

and to make them maximally correlated we maximize:

$$\begin{aligned} \underset{\mathbf{w}_{r,c}^j, \mathbf{w}_{d,c}^j}{maximize} \ & Corr(\mathbf{Y}_r^j, \mathbf{Y}_d^j) \\ & = Corr(\mathbf{w}_{r,c}^j \mathbf{X}_r, \mathbf{w}_{d,c}^j \mathbf{X}_d) \end{aligned} \tag{4.20}$$

in which superscript j refers to the j^{th} row of the corresponding matrices.

Canonical correlation analysis [33, 30] solves this analytically as an eigenproblem, in which each eigenvector gives one row of the projection and altogether provides the full projection matrices which lead to the maximum correlation between them.

Based on our intuition about insufficiency of correlated components for recognition tasks, in the second step, we fix correlation projections $(\mathbf{W}_{r,c}, \mathbf{W}_{d,c})$ and apply a reconstruction independent component analysis formulation [48], to extract modality-specific components for RGB and depth separately.

$$\mathbf{Z}_r = \mathbf{W}_{r,i}\mathbf{X}_r \qquad , \qquad \mathbf{Z}_d = \mathbf{W}_{d,i}\mathbf{X}_d \tag{4.21}$$

For RGB features we optimize:

$$\begin{aligned} &\underset{\mathbf{W}_{r,i}}{mininize} \ \frac{\lambda}{m} \left\| \widetilde{\mathbf{X}}_r - \mathbf{X}_r \right\|_F^2 + \sum_j \left\| \mathbf{W}_{r,i}^j \mathbf{X}_r \right\|_1 \\ &where\ \widetilde{\mathbf{X}}_r = \left[\mathbf{W}_{r,c}^T, \mathbf{W}_{r,i}^T \right] \begin{bmatrix} \mathbf{W}_{r,c} \\ \mathbf{W}_{r,i} \end{bmatrix} \mathbf{X}_r \end{aligned} \tag{4.22}$$

Similarly for depth features we optimize:

$$\begin{aligned} &\underset{\mathbf{W}_{d,i}}{mininize} \ \frac{\lambda}{m} \left\| \widetilde{\mathbf{X}}_d - \mathbf{X}_d \right\|_F^2 + \sum_j \left\| \mathbf{W}_{d,i}^j \mathbf{X}_d \right\|_1 \\ &where\ \widetilde{\mathbf{X}}_d = \left[\mathbf{W}_{d,c}^T, \mathbf{W}_{d,i}^T \right] \begin{bmatrix} \mathbf{W}_{d,c} \\ \mathbf{W}_{d,i} \end{bmatrix} \mathbf{X}_d \end{aligned} \tag{4.23}$$

Upon convergence of (4.22) and (4.23), the RGB+D features of each trajectory (k) can be represented as a quadruple: $\{\mathbf{Z}_r(k), \mathbf{Y}_r(k), \mathbf{Y}_d(k), \mathbf{Z}_d(k)\}$.

4.5 Experiments

This section presents our experimental setup and the results of the proposed methods on three RGB+D action recognition datasets. The description of the datasets are provided in section 2.3.1.

4.5.1 Experimental Setup

The proposed methods are evaluated on three RGB+D action recognition datasets. All these datasets are collected using the Microsoft Kinect sensor in an indoor environment.

Since the RGB and depth sequences are not fully aligned and not synced in Kinect sequences, convolutional cubes have to be large enough so that they mostly cover the same parts of the video between the two modalities. To apply the convolutional network, we consider four temporal quarters of the videos. In this way, each input sample has four temporal segments in our convolutional network and the factorized components of all these segments, together with holistic features of the entire sample are considered as the inputs of the stacked network.

To cover various aspects of RGB+D motion and appearances of input samples, we used a combination of different features. For depth channel, we extract histogram of oriented 4D normals (HON4D) [67], dynamic skeletons (DS), and dynamic depth patterns (DDP) [34]. From RGB videos, we extract dynamic color patterns (DCP) [34] and dense trajectory features [107]. In all of the experiments, different features of each modality are concatenated and then fed to a PCA to reduce their dimensions to a fixed size of 200 elements for local level and 400 for holistic level analysis.

The proper depth of the networks is found to be two layers for the local convolutional and three layers for the holistic network, via cross-validation. The size of $\mathbf{X}_{\cdot}$, $\mathbf{Y}_{\cdot}$, and $\mathbf{Z}_{\cdot}$ vectors is fixed as 100 for local features and 200 for holistic and stacked networks in our experiments.

In our experiments, the optimal values of the hyper-parameters in CIA aand SSLM are found via leave-one-sample-out cross-validation over training samples. Due to the difference of the datasets, these hyper-parameters are tuned for each dataset separately.

To show the effectiveness of our method, we compare it with two baseline methods below:

Baseline method 1: descriptor level fusion. In this method, we concatenate all the input RGB and depth-based features and train a linear SVM for classification.

Baseline method 2: kernel level combination. For this baseline method, we calculate the RBF kernel matrices based on all the input RGB and depth-based features and combine them linearly to classify in the form of multi-kernel SVM. We find the weights of kernels via a brute force search in a cross-validation setting using training samples [10].

In the following tables, we report the results of our method in two settings:

Ours Kernel is the kernel combination of the hierarchically factorized components of our correlation-independence analysis network. This is similar to the approach of baseline method 2, but using the factorized components of the deep CIA network for calculation of the kernel values.

Ours SSLM: refers to the proposed structured sparsity learning machine based on the hierarchically factorized components described in section 4.3.

4.5.2 Online RGB+D Action Dataset

Three different recognition scenarios are defined on this dataset. The first and second scenarios are cross-subject tests. In the first scenario, the first 8 actors are assigned for training and the second 8 actors are for testing. The samples of the second scenario are the same as the first one but training and testing samples are swapped. The third scenario is a cross-environment setting. The videos of the third 8 actors are collected in another location and are considered as test data. The other 16 actors' videos are used for training. The first and second scenarios are cross-subject and the third is a cross-environment evaluation.

Table 4.1 compares the results of the deep correlation-independence analysis (CIA) and structure sparsity learning machine (SSLM), and baseline methods on this dataset. The results of this experiment show our deep correlation-independence networks successfully decompose input features into a more powerful representation which leads into a clear improvement on the classification performance. They also show our SSLM can select the discriminative components and layers and learns a better classifier.

We also compare different structures of our deep correlation-independence analysis networks. For each scenario, we report the performance of three structures. "Holistic" refers to the 3-layer deep network applied on holistic features. "Local" is the 2-layer convolutional network applied on local features. "Stacked local+holistic" is the stacked local and holistic networks, as illustrated in Fig. 4.3. The results are reported in Table 4.2. We conclude that the local and holistic features are complementary and applying stacked local+holistic network can improve the final classification accuracy.

In our third experiment on this dataset, performance of the proposed networks is compared with a similar network without independent components. The reference network acts similarly to traditional CCA methods. We compare these two networks on the "local" network of third scenario. The result is shown in Table 4.3. We can see including independent components is beneficial and improves the accuracy. The second observation is our method improves the performance more significantly by having multiple layers. Without independent components, the values of correlated units can not change much, on higher layers. This shows our proposed structure is suitable for cascading more layers and decomposing the features layer by layer.

Table 4.4 compares our results with the state-of-the-art method on this dataset. Due to the recency of this dataset, only two other works reported results on this dataset. As shown, our method outperforms their results with a large margin, which demonstrates

Evaluation Dataset	Baseline Method 1	Baseline Method 2	Ours Kernel	Ours SSLM
Online RGB+D Action S1	86.6%	91.1%	92.9%	95.5%
Online RGB+D Action S2	85.6%	91.0%	91.9%	93.7%
Online RGB+D Action S3	73.0%	80.2%	82.0%	83.8%
MSR Daily Activity 3D	91.9%	94.4%	96.3%	97.5%
3D Action Pairs	97.7%	98.3%	100.0%	100.0%
RGBD-HuDaAct	95.1%	97.6%	98.3%	99.0%

Table 4.1 Comparison of the results of our methods with the baselines in all the three datasets: Online RGB+D Action, MSR Daily Activity 3D, 3D Action Pairs, and RGBD-HuDaAct datasets. S1, S2, and S3 refers to the three different scenarios of the Online RGB+D Action dataset. First column shows the performance of descriptor concatenation on all RGB+D input features. Second column reports the accuracy of the kernel combination on the same set of features. Third column shows the result of our correlation-independence analysis. It employs a kernel combination for classification. Last column reports the accuracy of proposed structured sparsity learning machine.

the importance of RGB+D fusion for action recognition as well as the effectiveness of our proposed method for this task.

4.5.3 MSR Daily Activity 3D Dataset

Results of the evaluation on this benchmark are reported in Tables 4.1 and 4.2.

Table 4.5 also shows the accuracy comparison between the proposed method and the state-of-the-art results reported on this benchmark, in which we outperform all the other methods with about two percentage points of accuracy. This shows our RGB+D analysis method can effectively improve the performance of the action recognition system.

4.5.4 3D Action Pairs Dataset

Table 4.6 compares the accuracies between the proposed framework and the state-of-the-art methods reported on this benchmark. Our method ties with BHIM [38]) and our MMMP method (chapter 3) in saturating the benchmark by achieving the flawless 100% accuracy on this dataset.

Evaluation Dataset	Network Structure	Ours Kernel	Ours SSLM
Online RGB+D Action S1	Holistic	90.2%	92.0%
Online RGB+D Action S1	Local	92.9%	93.8%
Online RGB+D Action S1	Stacked Local+Holistic	92.9%	95.5%
Online RGB+D Action S2	Holistic	87.4%	91.0%
Online RGB+D Action S2	Local	88.3%	89.2%
Online RGB+D Action S2	Stacked Local+Holistic	91.9%	93.7%
Online RGB+D Action S3	Holistic	79.3%	82.0%
Online RGB+D Action S3	Local	75.7%	77.5%
Online RGB+D Action S3	Stacked Local+Holistic	82.0%	83.8%
MSR Daily Activity 3D	Holistic	95.0%	96.3%
MSR Daily Activity 3D	Local	95.0%	96.9%
MSR Daily Activity 3D	Stacked Local+Holistic	96.3%	97.5%
3D Action Pairs	Holistic	98.9%	99.4%
3D Action Pairs	Local	99.4%	98.9%
3D Action Pairs	Stacked Local+Holistic	100.0%	100.0%
RGBD-HuDaAct	Holistic	98.3%	99.0%
RGBD-HuDaAct	Local	98.7%	98.7%
RGBD-HuDaAct	Stacked Local+Holistic	98.3%	99.0%

Table 4.2 Performance comparison for holistic network, local network, and stacked local+holistic (Fig. 4.3) networks on all the four datasets. Reported are the results of our method using kernel combination and SSLM.

Network	Layer 1 SSLM	2 Layers SSLM
Local Without **Z**	73.0%	73.9%
Local With **Z**	76.6%	77.5%

Table 4.3 Comparison with a correlation network (without independent components) on the Online RGB+D Action dataset, local network, scenario 3. Without **Z** components, the network is limited to the correlated ones and acts similar to CCA.

Methods	Setup	Accuracy
Orderlet [128]	Same environment	71.4%
Meng *et al.* [61]	Same environment	75.8%
Proposed Method	Same environment	**94.6%**
Orderlet [128]	Cross-environment	66.1%
Proposed Method	Cross-environment	**83.8%**

Table 4.4 Performance comparison of proposed multimodal correlation-independence analysis with the state-of-the-art results on Online RGB+D Action dataset. Same environment setup is the average of S1 and S2 scenarios, and cross-environment setup is the same as S3 scenario.

Method	Accuracy
HON4D [67]	80.0%
SSFF [84]	81.9%
ToSP [88]	84.4%
RGGP [56]	85.6%
Actionlet [110]	85.8%
SVN [125]	86.3%
BHIM [38]	86.9%
DCSF+Joint [120]	88.2%
MMTW [112]	88.8%
Depth Fusion [140]	88.8%
MMMP [83]	91.3%
DL-GSGC [60]	95.0%
JOULE-SVM [34]	95.0%
Range-Sample [58]	95.6%
Proposed Method	**97.5%**

Table 4.5 Performance comparison of the proposed multimodal correlation-independence analysis with the state-of-the-art methods on MSR Daily Activity dataset.

Method	Accuracy
DHOG [126]	66.11%
Actionlet [110]	82.22%
HON4D [67]	96.67%
MMTW [112]	97.22%
HOG3D-LLC [70]	98.33%
HOPC [73]	98.33%
SVN [125]	98.89%
BHIM [38]	100.0%
MMMP (chapter 3)	100.0%
Proposed Method	**100.0%**

Table 4.6 Performance comparison of proposed multimodal correlation-independence analysis with the state-of-the-art methods on 3D Action Pairs dataset.

4.5.5 Comparison with Single Modality

In Table 4.7, we compare our method with baseline method 2, based on single modality features. Since each modality also has holistic and multiple local features, we perform baseline kernel combination to produce the results. For a fair comparison, we use kernel combination for classification based on our factorized components. It is not surprising to observe that our method outperforms the baseline, since ours integrates RGB and depth information effectively.

4.5.6 Analysis of Factorized Components

Next is to study how much of information each of the factorized components in the proposed deep network hold. Table 4.8 shows the proportion of the weights assigned by SSLM to the factorized components of the stacked local+holistic networks. Reported values are the ℓ_2 norms of all the corresponding weights to each of the components, learned by SSLM on the stacked local+holistic networks. The assigned weights to $\mathbf{Y}^3$ are relatively high on all the evaluations, which supports our initial argument about robustness and discriminative properties of the correlated factorized components. The independent components of both the modalities in all three layers also gain weights, which shows they also carry informative features and are complementary for the action classification task.

Method	MSR Daily Activity 3D	3D Action Pairs	RGBD HuDa Act	Online RGBD Action S1	Online RGBD Action S2	Online RGBD Action S3
Baseline 2 on RGB-based Local+Holistic	89.4%	97.7%	95.2%	81.3%	85.6%	75.7%
Baseline 2 on depth-based Local+Holistic	92.5%	97.7%	79.1%	85.7%	84.7%	66.7%
Ours Kernel	**96.3%**	**100.0%**	**98.3%**	**92.9%**	**91.9%**	**82.0%**

Table 4.7 Comparison between our method and baseline method 2 on single modality RGB and depth based input features, on all the datasets.

Dataset	$\mathbf{Z}_r^1$	$\mathbf{Z}_r^2$	$\mathbf{Z}_r^3$	$\mathbf{Y}^3$	$\mathbf{Z}_d^3$	$\mathbf{Z}_d^2$	$\mathbf{Z}_d^1$
Online S1	0.12	0.13	0.18	0.20	0.13	0.05	0.18
Online S2	0.29	0.06	0.03	0.42	0.06	0.11	0.03
Online S3	0.14	0.12	0.06	0.26	0.13	0.00	0.28
Daily	0.17	0.10	0.09	0.19	0.15	0.11	0.19
Pairs	0.06	0.02	0.16	0.42	0.01	0.03	0.29

Table 4.8 Proportion of the weights to factorized components in the learned SSLM classifier, for all the three datasets. Reported values are the ℓ_2 norms of all the corresponding weights to each of the components, learned by SSLM on the stacked local+holistic networks.

4.5.7 RGBD-HuDaAct Dataset

In our experiments for this dataset, we follow the evaluation setup described in [63].

Atomic Local Level Feature Analysis

Unlike most of the other datasets, this benchmark provides fully synchronized and aligned set of RGB and depth videos. This important characteristic enables us to apply the atomic level of analysis on local RGB and depth features within the video samples.

As our atomic local level features, we extract the tracked dense trajectories [107] in RGB sequences and their HOG, HOF, MBHX, and MBHY descriptors from both modalities.

To evaluate the effectiveness of the proposed RGB+D analysis, we apply a single layer CIA to decompose RGB and depth descriptors of the trajectories to their correlated and independent components. For training stage, we sample a set of 40K trajectories from training set. The output of the analysis, which are four factorized components for each trajectory are clustered separately by K-Means with codebook size 1K. LLC coding [113] and BOF framework are applied on the codes of all the trajectories from each RGB+D video sample to extract their global representations.

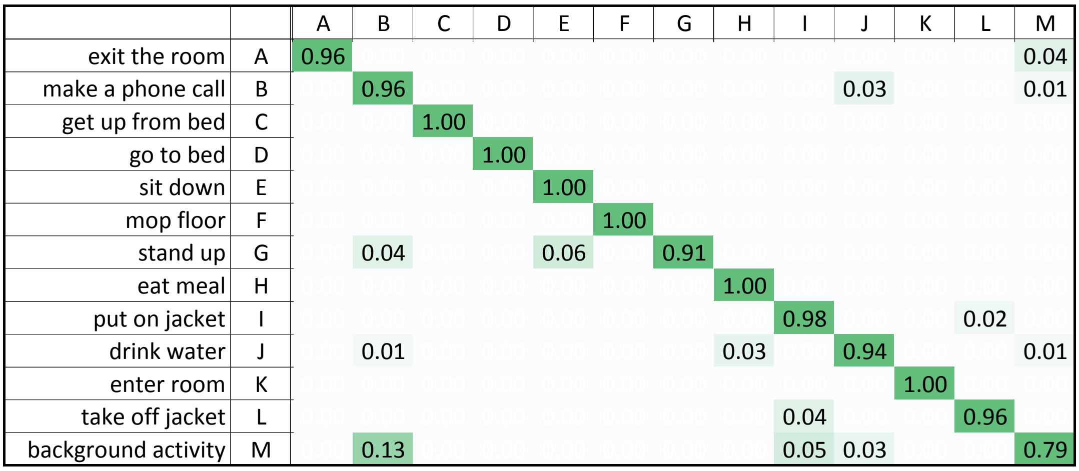
In the final step, a linear SVM is used as the action classifier trained on the extracted global representations of the action video samples.

We evaluated the performance of canonical correlation analysis (CCA) method also. In our implementation of the CCA-RICA method (section 4.4) we used the provided codes by the authors of [9] for CCA and [48] for RICA.

Table 4.9 shows the results of all the experiments described in this section and compares them with other state-of-the-art methods.

At first, we evaluated the performance of correlated components of CCA without any modality specific features, which achieves 93.9% outperforming all the reported results on this benchmark. Compared to the accuracy of RGB+D linear coding [53], which has the most similar pipeline of action recognition to ours, CCA components shows about two percents improvement. This approves the robustness of correlated components and their advantage over using a simple combination of features from the two modalities.

In the next step, we apply RICA to extract modality-specific components for RGB and depth local features. Adding independent components improves the accuracy of the classification by 2.5 more percents. This supports our argument about the importance of modality-specific components and their discriminative strengths for action classification.



		A	B	C	D	E	F	G	H	I	J	K	L	M
exit the room	A	0.96												0.04
make a phone call	B		0.96								0.03			0.01
get up from bed	C			1.00										
go to bed	D				1.00									
sit down	E					1.00								
mop floor	F						1.00							
stand up	G		0.04			0.06		0.91						
eat meal	H								1.00					
put on jacket	I									0.98			0.02	
drink water	J		0.01						0.03		0.94			0.01
enter room	K											1.00		
take off jacket	L									0.04			0.96	
background activity	M		0.13							0.05	0.03			0.79

Fig. 4.4 Confusion matrix for CCA-RICA method on atomic local level features RGBD-HuDaAct dataset. Ground truth action labels are on rows and detections are on columns of the grid.

The confusion matrix for this method is illustrated in Fig. 4.4. The majority of the misclassification are caused by the background activity class. This class contains samples of random motion and other simple activities which are not covered by other 12 classes, like walking around or stay seated without much of motion. Therefore it is inevitable to have some confusion between this class with classes which contain very small amount of clear motion *e.g.* making a phone call. Similar action classes with reverse temporal order are also mixed up, *e.g.* sit down and stand up, or put on jacket and take off jacket classes have the same appearance within individual frames, and their only differences are the arrangement of frames over time.

Next, we evaluate the CIA method on this atomic local level. Since the extraction of correlated and independent components are done together in this method, optimization of the overall cost function can effectively discover a proper range of the correlated components between RGB and depth and assign the remaining information inside the input features as independent components. We believe this is the reason CIA outperforms all other techniques by performing 97.9% of correct classification and achieves the state-of-the-art accuracy on this dataset. Compared to CCA-RICA method, CIA improves the error rate by more than 40% which is a notable improvement. The confusion matrix of this experiment is also reported in Fig. 4.5. Compared to the mixed up cases of the CCA-RICA method (Fig. 4.4), the confusion patterns are similar but furthered improved.

		A	B	C	D	E	F	G	H	I	J	K	L	M
exit the room	A	0.98	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02
make a phone call	B	0.00	0.97	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00
get up from bed	C	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
go to bed	D	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
sit down	E	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
mop floor	F	0.00	0.00	0.00	0.00	0.00	0.98	0.00	0.00	0.00	0.00	0.00	0.00	0.02
stand up	G	0.00	0.02	0.00	0.00	0.00	0.00	0.98	0.00	0.00	0.00	0.00	0.00	0.00
eat meal	H	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.98	0.00	0.02	0.00	0.00	0.00
put on jacket	I	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
drink water	J	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.96	0.00	0.00	0.01
enter room	K	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00
take off jacket	L	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.96	0.00
background activity	M	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.03	0.00	0.00	0.92

Fig. 4.5 Confusion matrix for CIA method on atomic local level features of RGBD-HuDaAct dataset. Ground truth action labels are on rows and detections are on columns of the grid.

Global Level Feature Analysis

Similar to other three datasets reported in this chapter, we perform the proposed RGB+D analysis on the global representations extracted from input samples. For RGB signals, the features are HOG, HOF, MBHX, and MBHY descriptors of dense trajectories [107], followed by a K-means clustering and locality-constrained linear coding (LLC) [113] to calculate their global representations as bags-of-features. For depth, we extract HON4D features [67] for holistic and local depth based features. The results of this experiment are reported in Tables 4.9, 4.1, 4.2, and 4.7 in a similar evaluation setup to other datasets.

As can be seen in Table 4.9, applying CIA analysis in a deep and stacked framework outperforms all the current methods as well as the atomic local level analysis, and achieved the outstanding performance of 99.0% on this benchmark, which shows more than 50% improvement on the error rate compared to the atomic local level CIA analysis.

Other reported results are also in accord with our results on other datasets and approve our arguments about the effectiveness of the structure of the proposed framework in this chapter.

4.6 Chapter Summary

This chapter presents a new deep learning framework for a hierarchical correlated-independent component factorization, to analyze RGB+D features of human action

Method	**Accuracy**
3D-MHIs [63]	70.5%
iM^{2}EDM [15]	76.8%
MF-HMM [40]	78.6%
DLMC-STIPs [63]	81.5%
DIMC-STIPs [137]	87.7%
STIP HOGHOF+LDP [136]	89.1%
Part-based BOW-Pyramid [94]	91.7%
RGB+D Linear Coding [53]	92.0%
CCA (Atomic Level)	93.9%
CCA-RICA (Atomic Level)	96.4%
Single Unit CIA (Atomic Level)	**97.9%**
Deep Stacked CIA (Global Level)	**99.0%**

Table 4.9 Performance Comparison on RGBD-HuDaAct Dataset

videos. Each layer of the proposed network is an autoencoder based component factorization unit, which decomposes its multimodal input features into correlated and independent parts. We further extended our deep factorization framework by applying it in a convolutional setting.

In addition, we proposed a structured sparsity based classifier which utilizes mixed norms to apply component and layer selection for a proper fusion of decomposed feature components.

Provided experimental results on four RGB+D action recognition datasets show the strength of our deep correlation-independence analysis and the proposed structured sparsity learning machine by achieving the state-of-the-art performances on all the reported benchmarks.

Chapter 5

NTU-RGB+D: A Large Scale Dataset for 3D Human Activity Analysis

During our research and experiments on depth-based and RGB+D based human activity analysis in previous chapters, we found a major bottleneck of the research in this direction: lack of enough data. To the best of our knowledge, all the current RGB+D human action datasets were highly limited in various aspects. These limitations prevents us from moving towards more complex learning frameworks.

This urged us to collect and propose a new large-sized RGB+D human actions dataset. In this chapter, we introduce our dataset and propose a novel data-driven learning framework for action recognition, trained and evaluated on this dataset.

5.1 Introduction

Unlike the RGB-based counterpart, 3D video analysis suffers from the lack of large-sized benchmark datasets. Yet there are no sources of publicly shared 3D videos such as the YouTube to supply "in-the-wild" samples. This limits our ability to build large-sized benchmarks to evaluate and compare the strengths of different methods, especially the recent data-hungry techniques like deep learning. To the best of our knowledge, all the current 3D action recognition benchmarks have limitations in various aspects.

First is the small number of subjects and very narrow range of performers' ages, which makes the intra-class variation of the actions very limited. The constitution of human activities depends on the age, gender, culture and even physical conditions of the subjects. Therefore, variation of human subjects is crucial for an action recognition benchmark.

The second factor is the number of action classes. When only a very small number of classes are available, each action class can be easily distinguished by finding a simple motion pattern or even the appearance of an interacted object. But when the number of classes grows, the motion patterns and interacting objects will be shared between classes and classifying them would be more challenging.

Third is the highly restricted camera views. For most of the datasets, all the samples are captured from a front view with a fixed camera viewpoint. For some others, views are bounded to fixed front and side views, using multiple cameras at the same time.

Finally and most importantly, the highly limited number of video samples prevents us from applying the most advanced data-driven learning methods to this problem. Although some attempts have been done to do so [22, 117], they suffered from overfitting and had to scale down the size of learning parameters; as a result, they clearly need many more samples to generalize and perform better on testing data.

To overcome these limitations, we develop a new large-scale benchmark dataset for 3D human activity analysis. The proposed dataset consists of $56,880$ RGB+D video samples, captured from 40 different human subjects, using Microsoft Kinect v2. We have collected RGB videos, depth sequences, skeleton data (3D locations of 25 major body joints), and infrared frames. Samples are captured in 80 distinct camera viewpoints. The age range of the subjects in our dataset is from 10 to 35 years which brings more realistic variation to the quality of actions. Although our dataset is limited to indoor scenes, due to the operational limitation of the acquisition sensor, we provide the ambiance inconstancy by capturing in various background conditions. This large amount of variation in subjects and views makes it possible to have more accurate cross-subject and cross-view evaluations for various 3D-based action analysis methods. Table 5.1 shows the comparison between our dataset and other currently existing datasets for depth-based action recognition. In almost all of the quantitative aspects, we provide larger size of data in orders of magnitude.

The proposed dataset can help the community to move forward in 3D human activity analysis and make it possible to apply data-hungry methods such as deep learning techniques for this task.

As another contribution, inspired by the physical characteristics of human body motion, we propose a novel part-aware extension of the long short-term memory (LSTM) model [32]. Human actions can be interpreted as interactions of different parts of the body. In this way, the joints of each body part always move together and the combination of their 3D trajectories form more complex motion patterns. By splitting the memory cell of the LSTM into part-based sub-cells, the recurrent network will learn the long-term patterns specifically for body parts and the output of the unit will be learned from all the sub-cells.

Our experimental results on the proposed dataset shows the clear advantages of data-driven learning methods over state-of-the-art hand-crafted features.

The rest of this chapter is organized as follows: Section 5.2 introduces the proposed dataset, its structure, and defined evaluation criteria. Section 5.3 presents our new part-aware long short-term memory network for action analysis in a recurrent neural network fashion. Section 5.4 shows the experimental evaluations of state-of-the-art hand-crafted features as well as the proposed recurrent learning method on our benchmark, and section 5.5 concludes the chapter.

Datasets		Samples	Classes	Subjects	Views	Sensor	RGB	Depth	Joints	Other	Year
MSR-Action3D	[51]	567	20	10	1	N/A		✓	✓		2010
CAD-60	[90]	60	12	4	-	Kinect 360	✓	✓	✓		2011
RGBD-HuDaAct	[63]	1189	13	30	1	Kinect 360	✓	✓			2011
MSRDailyActivity3D	[109]	320	16	10	1	Kinect 360	✓	✓	✓		2012
ACT4^2	[18]	6844	14	24	4	Kinect 360	✓	✓			2012
UTKincet	[121]	200	10	10	-	Kincet 360	✓	✓	✓		2012
SBU	[131]	300	8	7	1	Kinect 360	✓	✓	✓		2012
CAD-120	[39]	120	20	4	-	Kinect 360	✓	✓	✓		2013
3D Action Pairs	[67]	360	12	10	1	Kinect 360	✓	✓	✓		2013
Multiview 3D Event	[118]	3815	8	8	3	Kinect 360	✓	✓	✓		2013
Online RGB+D Action	[128]	336	7	24	1	Kinect 360	✓	✓	✓		2014
G3Di	[7]	72	6	12	1	Kinect 360	✓	✓	✓		2014
LIRIS Human Activ.	[119]	828	10	21	-	Kinect 360	✓	✓			2014
Office Activity	[114]	1180	20	5	-	Kinect 360	✓	✓			2014
Northwestern-UCLA	[111]	1475	10	10	3	Kinect 360	✓	✓	✓		2014
UWA3D Multiview	[73]	$\sim$900	30	10	1	Kinect 360	✓	✓	✓		2014
Office Activity	[115]	1180	20	10	3	Kinect 360	✓	✓			2014
UTD-MHAD	[16]	861	27	8	1	Kinect 360+WIS	✓	✓	✓	ID	2015
UWA3D Multiview II	[71]	1075	30	10	5	Kinect 360	✓	✓	✓		2015
Proposed dataset (NTU RGB+D)		**56880**	**60**	**40**	**80**	**Kinect v.2**	✓	✓	✓	**IR**	**2016**

Table 5.1 Comparison between the developed dataset and other publicly available datasets for 3D action recognition. Our dataset provides many more samples, action classes, human subjects, and camera views in comparison with other available datasets for RGB+D action recogniton.

5.2 NTU-RGB+D Action Dataset

This section introduces the details and the evaluation criteria of the proposed RGB+D action recognition dataset.

5.2.1 The Structure of the Dataset

Data Modalities: To collect this dataset, we utilized Microsoft Kinect v2 sensors. We collected four different modalities of data from this sensor, which are depth-map sequences, RGB videos, IR frames, and skeletal data. Depth maps are sequences of two dimensional depth values in millimeters. To maintain all the information, we applied lossless compression for each individual frame. The resolution of each depth frame is 512×424. Joint information consists of 3-dimensional locations of 25 major body joints for detected and tracked human bodies in the scene. The corresponding pixels on RGB frames and depth maps are also provided for each joint and every frame. The configuration of body joints is illustrated in Fig. 5.2. RGB videos are recorded in the provided resolution of 1920×1080. Infrared sequences are also collected and stored frame by frame in 512×424.

Action Classes: We have 60 action classes in total, which are divided into three major groups.

Daily actions: 1-drinking, 2-eating, 3-brushing teeth, 4-brushing hair, 5-dropping, 6-picking up, 7-throwing, 8-sitting down, 9-standing up (from sitting position), 10-clapping, 11-reading, 12-writing, 13-tearing up paper, 14-wearing jacket, 15-taking off jacket, 16-wearing a shoe, 17-taking off a shoe, 18-wearing on glasses, 19-taking off glasses, 20-puting on a hat/cap, 21-taking off a hat/cap, 22-cheering up, 23-hand waving, 24-kicking something, 25-reaching into self pocket, 26-hopping, 27-jumping up, 28-making/answering a phone call, 29-playing with phone, 30-typing, 31-pointing to something, 32-taking selfie, 33-checking time (on watch), 34-rubbing two hands together, 35-bowing, 36-shaking head, 37-wiping face, 38-saluting, 39-putting palms together, 40-crossing hands in front.

Medical actions: 41-sneezing/coughing, 42-staggering, 43-falling down, 44-touching head (headache), 45-touching chest (stomachache/heart pain), 46-touching back (back-pain), 47-touching neck (neck-ache), 48-vomiting, 49-fanning self.

Mutual actions: 50-punching/slapping other person, 51-kicking other person, 52-pushing other person, 53-patting other's back, 54-pointing to the other person, 55-hugging, 56-giving something to other person, 57-touching other person's pocket, 58-handshaking, 59-walking towards each other, 60-walking apart from each other.

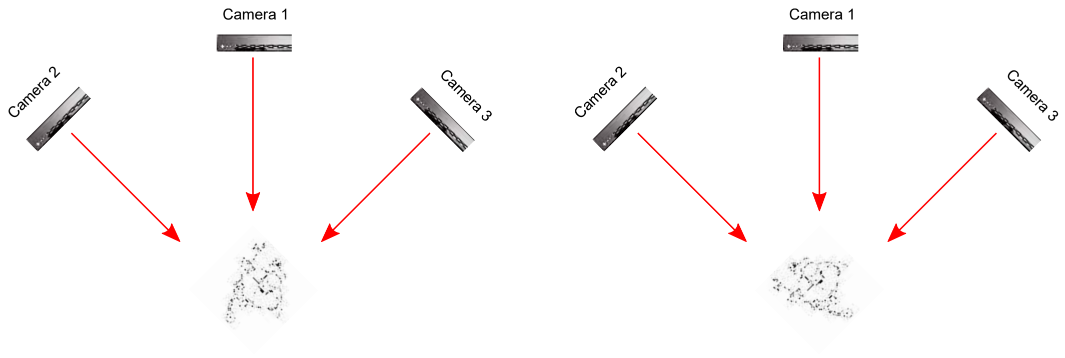


Fig. 5.1 The arrangement of the three Kinect cameras at the same time. For each setup, the three cameras were located at the same height but from three different horizontal viewing angles: front view, 45 degrees view, and side view, and each subject was asked to perform each action twice, once towards the left camera and once towards the right camera. Overall, we collect two front views, one left side view, one right side view, one left side 45 degrees view, and one right side 45 degrees view.

Subjects: We invited 40 distinct subjects for our data collection. The ages of the subjects are between 10 and 35. Fig. 5.5 shows the variety of the subjects in age, gender, and height. Each subject is assigned a consistent ID number over the entire dataset.

Views: We used three cameras at the same time to capture three different views from the same action. In this way, we capture two front views, one left side view, one right side view, one left side 45 degrees view, and one right side 45 degrees view. The three cameras are assigned consistent camera numbers. Camera 1 always sees the 45 degrees views, while camera 2 and 3 see front and side views. This is illustrated in Fig. 5.1.

To further increase the camera views, on each setup we changed the height and distances of the cameras to the subjects, as reported in Table 5.2. All the camera and setup numbers are annotated for each video sample.

5.2.2 Benchmark Evaluations

To have standard evaluations for all the reported results on this benchmark, we define precise criteria for two types of action classification evaluation, as described in this section. For each of these two, we report the classification accuracy in percentage.

Setup No.	Height (m)	Distance (m)	Setup No.	Height (m)	Distance (m)
1	1.7	3.5	2	1.7	2.5
3	1.4	2.5	4	1.2	3.0
5	1.2	3.0	6	0.8	3.5
7	0.5	4.5	8	1.4	3.5
9	0.8	2.0	10	1.8	3.0
11	1.9	3.0	12	2.0	3.0
13	2.1	3.0	14	2.2	3.0
15	2.3	3.5	16	2.7	3.5
17	2.5	3.0			

Table 5.2 Height and distance of the three cameras for each collection setup. All height and distance values are in meters.

Cross-Subject Evaluation

In cross-subject evaluation, we split the 40 subjects into training and testing groups. Each group consists of 20 subjects. For this evaluation, the training and testing sets have $40,320$ and $16,560$ samples, respectively. The IDs of training subjects in this evaluation are: 1, 2, 4, 5, 8, 9, 13, 14, 15, 16, 17, 18, 19, 25, 27, 28, 31, 34, 35, 38; remaining subjects are reserved for testing.

Cross-View Evaluation

For cross-view evaluation, we pick all the samples of camera 1 for testing and samples of cameras 2 and 3 for training. In other words, the training set consists of front and two side views of the actions, while testing set includes left and right 45 degree views of the action performances. For this evaluation, the training and testing sets have $37,920$ and $18,960$ samples, respectively.

5.3 Part-Aware LSTM Network

In this section, we introduce a new data-driven learning method to model the human actions using our collected 3D action sequences.

Human actions can be interpreted as time series of body configurations. These body configurations can be effectively and succinctly represented by the 3D locations of major joints of the body. In this fashion, each video sample can be modeled as a sequential representation of configurations.

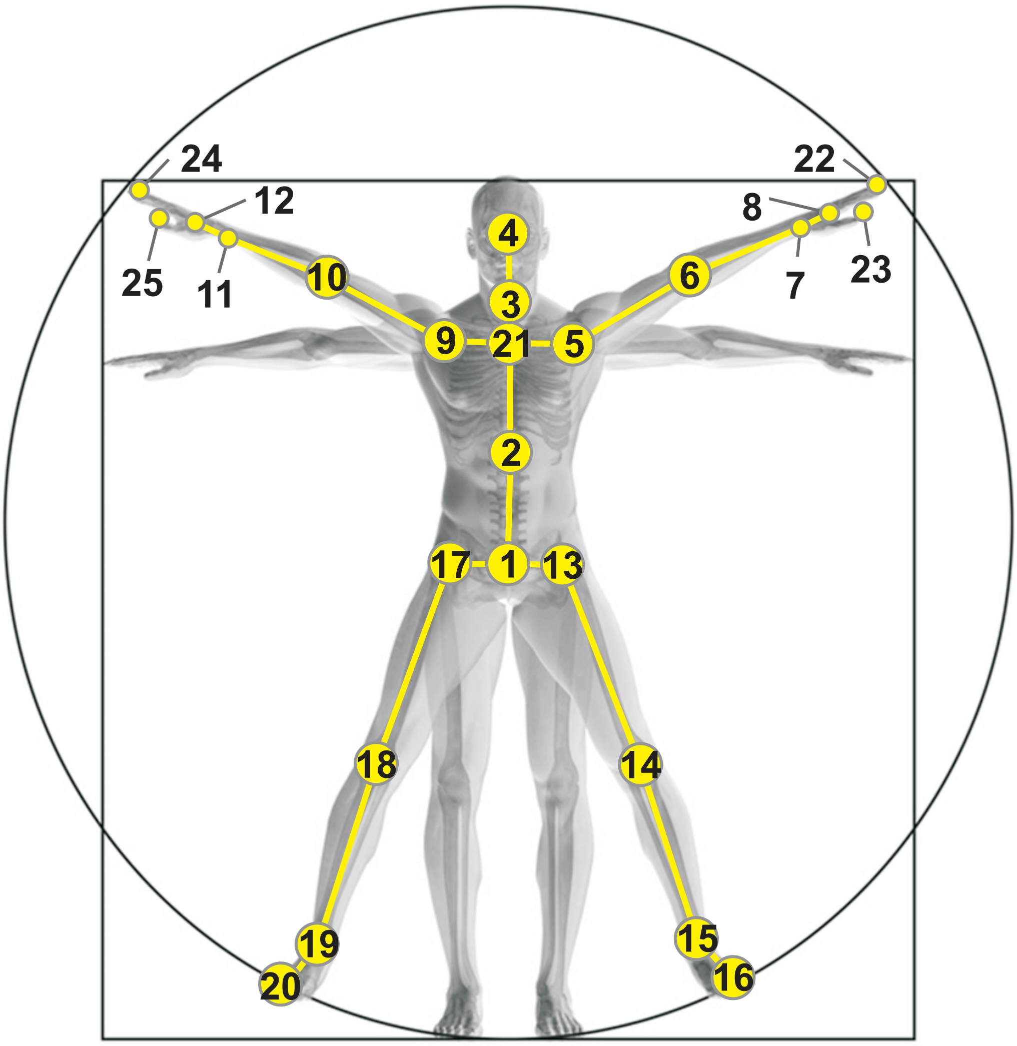


Fig. 5.2 Configuration of 25 body joints in our dataset. The labels of the joints are: 1-base of the spine 2-middle of the spine 3-neck 4-head 5-left shoulder 6-left elbow 7-left wrist 8-left hand 9-right shoulder 10-right elbow 11-right wrist 12-right hand 13-left hip 14-left knee 15-left ankle 16-left foot 17-right hip 18-right knee 19-right ankle 20-right foot 21-spine 22-tip of the left hand 23-left thumb 24-tip of the right hand 25-right thumb

Recurrent Neural Networks (RNNs) and Long Short-Term Memory Networks (LSTMs) [32] have been shown to be among the most successful deep learning models to encode and learn sequential data in various applications [91, 21, 11, 36].

In this section, we introduce the traditional recurrent neural networks and then propose our part-aware LSTM model.

5.3.1 Traditional RNN and LSTM

A recurrent neural network transforms an input sequence $(\mathbf{X})$ to another sequence $(\mathbf{Y})$ by updating its internal state representation $(\mathbf{h}_t)$ at each time step (t) as a linear function of the last step's state and the input at the current step, followed by a nonlinear scaling function. Mathematically:

$$\mathbf{h}_t = \sigma \left(\mathbf{W} \begin{pmatrix} \mathbf{x}_t \\ \mathbf{h}_{t-1} \end{pmatrix} \right) \tag{5.1}$$

$$\mathbf{y}_t = \sigma \left(\mathbf{V}\mathbf{h}_t \right) \tag{5.2}$$

where $t \in \{1,..,T\}$ represents time steps, and $\sigma \in \{Sigm, Tanh\}$ is a nonlinear scaling function.

Layers of RNNs can be stacked to build a deep recurrent network:

$$\mathbf{h}_t^l = \sigma \left(\mathbf{W}^l \begin{pmatrix} \mathbf{h}_t^{l-1} \\ \mathbf{h}_{t-1}^l \end{pmatrix} \right) \tag{5.3}$$

$$\mathbf{h}_t^0 := \mathbf{x}_t \tag{5.4}$$

$$\mathbf{y}_t = \sigma \left(\mathbf{V}\mathbf{h}_t^L \right) \tag{5.5}$$

where $l \in \{1,...,L\}$ represents layers.

Traditional RNNs have limited abilities to keep long-term representation of the sequences and were unable to discover relations among long-ranges of inputs. To alleviate this drawback, Long Short-Term Memory Network [32] was introduced to keep a long term memory inside each RNN unit and learn when to remember or forget information stored inside its internal memory cell (c^t):

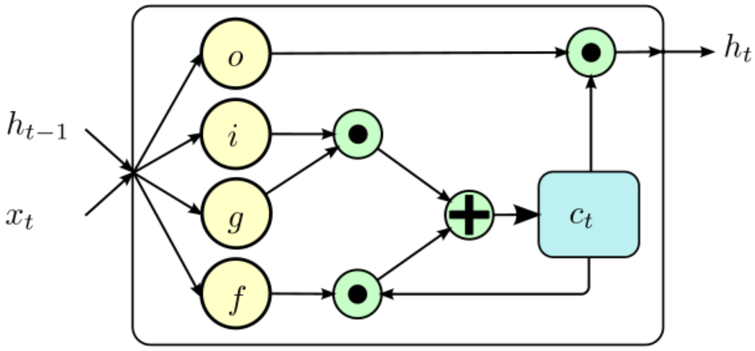


Fig. 5.3 Schema of a long short-term memory (LSTM) unit. o is the output gate, i is the input gate, g is the input modulation gate, and f is the forget gate. c is the memory cell to keep the long term context.

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} Sigm \\ Sigm \\ Sigm \\ Tanh \end{pmatrix} \left(\mathbf{W} \begin{pmatrix} \mathbf{x}_t \\ \mathbf{h}_{t-1} \end{pmatrix} \right) \tag{5.6}$$

$$c_t = f \odot c_{t-1} + i \odot g \tag{5.7}$$

$$h_t = o \odot Tanh(c_t) \tag{5.8}$$

In this model, i, f, o, and g denote input gate, forget gate, output gate, and input modulation gate respectively. Operator $\odot$ denotes element-wise multiplication. Fig. 5.3 shows the schema of this recurrent unit.

The output $\mathbf{y}_t$ is fed to a softmax layer to transform the output codes to probability values of class labels. To train such networks for action recognition, we fix the training output label for each input sample over time.

5.3.2 Proposed Part-Aware LSTM

In human actions, body joints move together in groups. Each group can be assigned to a major part of the body, and actions can be interpreted based on the interactions between body parts or with other objects. Based on this intuition, we propose a part-aware LSTM human action learning model. We dub the method P-LSTM.

Instead of keeping a long-term memory of the entire body's motion in the cell, we split it to part-based cells. It is intuitive and more efficient to keep the context of each body part independently and represent the output of the P-LSTM unit as a combination of independent body part context information. In this fashion, each part's cell has

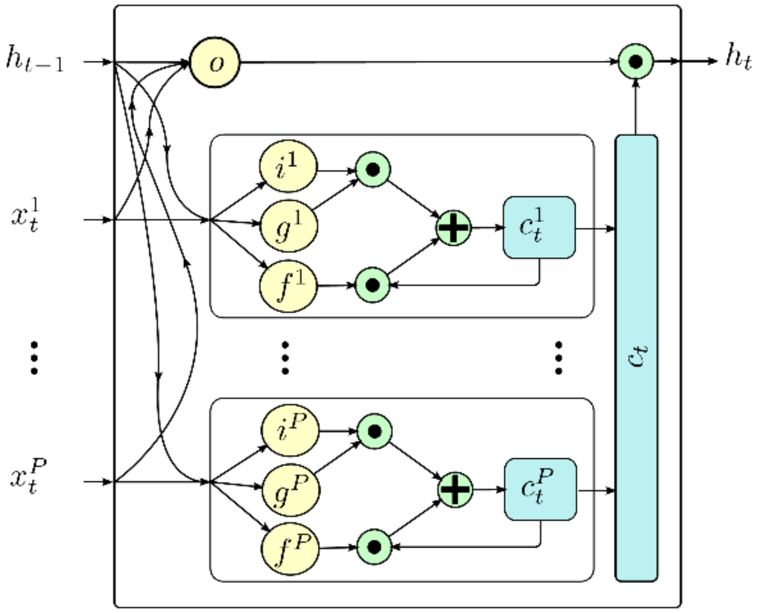


Fig. 5.4 Illustration of the proposed part-aware long short-term memory (P-LSTM) unit.

its individual input, forget, and modulation gates, but the output gate will be shared among the body parts. In our model, we group the body joints into five part groups: torso, two hands, and two legs.

At each frame t, we concatenate the 3D coordinates of the joints inside each part $p \in \{1,...,P\}$ and consider them as the input representation of that part, denoted as $\mathbf{x}_t^p$.

Thusly, the proposed P-LSTM is modeled as:

$$\begin{pmatrix} i^p \\ f^p \\ g^p \end{pmatrix} = \begin{pmatrix} Sigm \\ Sigm \\ Tanh \end{pmatrix} \left(\mathbf{W}^p \begin{pmatrix} \mathbf{x}_t^p \\ \mathbf{h}_{t-1} \end{pmatrix} \right) \tag{5.9}$$

$$c_t^p = f^p \odot c_{t-1}^p + i^p \odot g^p \tag{5.10}$$

$$o = Sigm \left(\mathbf{W}_o \begin{pmatrix} \mathbf{x}_t^1 \\ \vdots \\ \mathbf{x}_t^P \\ \mathbf{h}_{t-1} \end{pmatrix} \right) \tag{5.11}$$

$$h_t = o \odot Tanh \begin{pmatrix} \mathbf{c}_t^1 \\ \vdots \\ \mathbf{c}_t^P \end{pmatrix} \tag{5.12}$$

A graphical representation of the propsed P-LSTM is illustrated in Fig. 5.4.

Our P-LSTM learns the common temporal patterns of the parts independently and combines them in the global level representation for action recognition.

5.4 Experiments

In our experiments, we evaluate state-of-the-art depth-based action recognition methods and compare them with RNN, LSTM, and the proposed P-LSTM based on the evaluation criteria of our dataset.

5.4.1 Experimental Setup

We use the publicly available implementation of six depth-based action recognition methods and apply them on our new dataset benchmark. Among them, HOG^2 [65], Super Normal Vector [125], and HON4D [67] extract features directly from depth maps without using the skeletal information. Lie group [100], Skeletal Quads [24], and FTP Dynamic Skeletons [34] are skeleton based methods.

The other evaluated methods are RNN, LSTM, and the proposed P-LSTM method.

For skeletal representation, we apply a normalization preprocessing step. The original 3D locations of the body joints are provided in camera coordinate system. We translate them to the body coordinate system with its origin on the "middle of the spine" joint (number 2 in Fig. 5.2), followed by a 3D rotation to fix the X axis parallel to the 3D vector from "right shoulder" to "left shoulder", and Y axis towards the 3D vector from "spine base" to "spine". The Z axis is fixed as the new $X \times Y$. In the last step of normalization, we scale all the 3D points based on the distance between "spine base" and "spine" joints.

In the cases of having more than one body in the scene, we transform all of them with regard to the main actor's skeleton. To choose the main actor among the available skeletons, we pick the one with the highest amount of 3D body motion.

Kinect's body tracker is prone to detecting some objects *e.g.* seats or tables as bodies. To filter out these noisy detections, for each tracked skeleton we calculate the spread of the joint locations towards image axis and filtered out the ones whose X spread were more than 0.8 of their Y spread.

For our recurrent model evaluation, we reserve about five percent of the training data as validation set. The networks are trained on a large number of iterations and we pick the network with the least validation error among all the iterations and report its performance on testing data.

Method	Cross-Subject Accuracy	Cross-View Accuracy
HOG^2 [65]	32.24%	22.27%
Super Normal Vector [125]	24.56%	13.61%
HON4D [67]	30.56%	07.26%
Lie Group [100]	50.08%	52.76%
Skeletal Quads [24]	38.62%	41.36%
FTP Dynamic Skeletons [34]	60.23%	65.22%
1 Layer RNN	56.02%	60.24%
2 Layer RNN	56.29%	64.09%
1 Layer LSTM	59.14%	66.81%
2 Layer LSTM	60.69%	67.29%
1 Layer P-LSTM	62.05%	69.40%
2 Layer P-LSTM	**62.93%**	**70.27%**

Table 5.3 The results of the two evaluation settings of our benchmark using different methods. First three rows are depth-map based baseline methods. Rows 4, 5, and 6 are three skeleton based baseline methods. Following rows report the performance of RNN, LSTM and the proposed P-LSTM model. Our P-LSTM learning model outperforms other methods on both of the evaluation settings.

For each video sample at each training iteration, we split the video to $T = 8$ equal sized temporal segments and randomly pick one frame from each segment to feed the skeletal information of that frame as input to the recurrent leaning models in $t \in \{1, ..., T\}$ time steps.

For the baseline methods which use SVM as their classifier, to be able to manage the large scale of the data, we use Libliner SVM toolbox [25].

Our RNN, LSTM, and P-LSTM implementations are done on the Torch toolbox platform [19]. We use a Nvidia Tesla K40 GPU to run our experiments. In average, our training and testing takes about 50 and 2 milliseconds per sample on the above-mentioned platform.

5.4.2 Experimental Evaluations

The results of our evaluations of the above-mentioned methods are reported in Table 5.3. First three rows show the accuracies of the evaluated depth-map features. They perform better in cross-subject evaluation compared to the cross-view one. The reason for this difference is that in the cross-view scenario, the depth appearance of the actions are different and these methods are more prone to learning the appearances or view-dependent motion patterns.

Skeletal based features (Lie group [100], Skeletal Quads [24], and FTP Dynamic Skeletons [34]), perform better with a notable gap on both settings. They are stronger to generalize between the views because the 3D skeletal representation is view-invariant in essence, but it's prone to errors of the body tracker.

At the next step, we evaluate the discussed recurrent networks on this benchmark. Although RNN has the limitation in discovering long-term interdependency of inputs, they perform competitively with the hand-crafted methods. Stacking one more RNN layer improves the overall performance of the network, especially in cross-view scenario.

By utilizing long-term context in LSTM, the performances are improved significantly. LSTM's performance improves slightly by stacking one more layer.

At the last step, we evaluate the proposed P-LSTM model. By isolating the context memory of each body part and training the classifier based on their combination, we model a new way of regularization in the learning process of LSTM parameters. It utilizes the high intra-part and low inter-part correlation of input features to improve the learning process of the LSTM network. As shown in Table 5.3 P-LSTM outperforms all other methods by achieving 62.93% in cross-subject, and 70.27% in cross-view evaluation scenarios.

5.5 Chapter Summary

A large-scale RGB+D action recognition dataset is introduced in this chapter. Our dataset includes 56880 video samples collected from 60 action classes in highly variant camera settings. Compared to the current datasets for this task, our dataset is larger in orders and contains much more variety in different aspects.

The large scale of the collected data enables us to apply data-driven learning methods like Long Short-Term Memory networks in this problem and achieve better performance accuracies compared to hand-crafted features.

We also propose a Part-aware LSTM model to utilize the physical structure of the human body to further improve the performance of the LSTM learning framework.

The provided experimental results show the availability of large-scale data enables the data-driven learning frameworks to outperform hand-crafted features. They also show the effectiveness of the proposed P-LSTM model over traditional recurrent models.

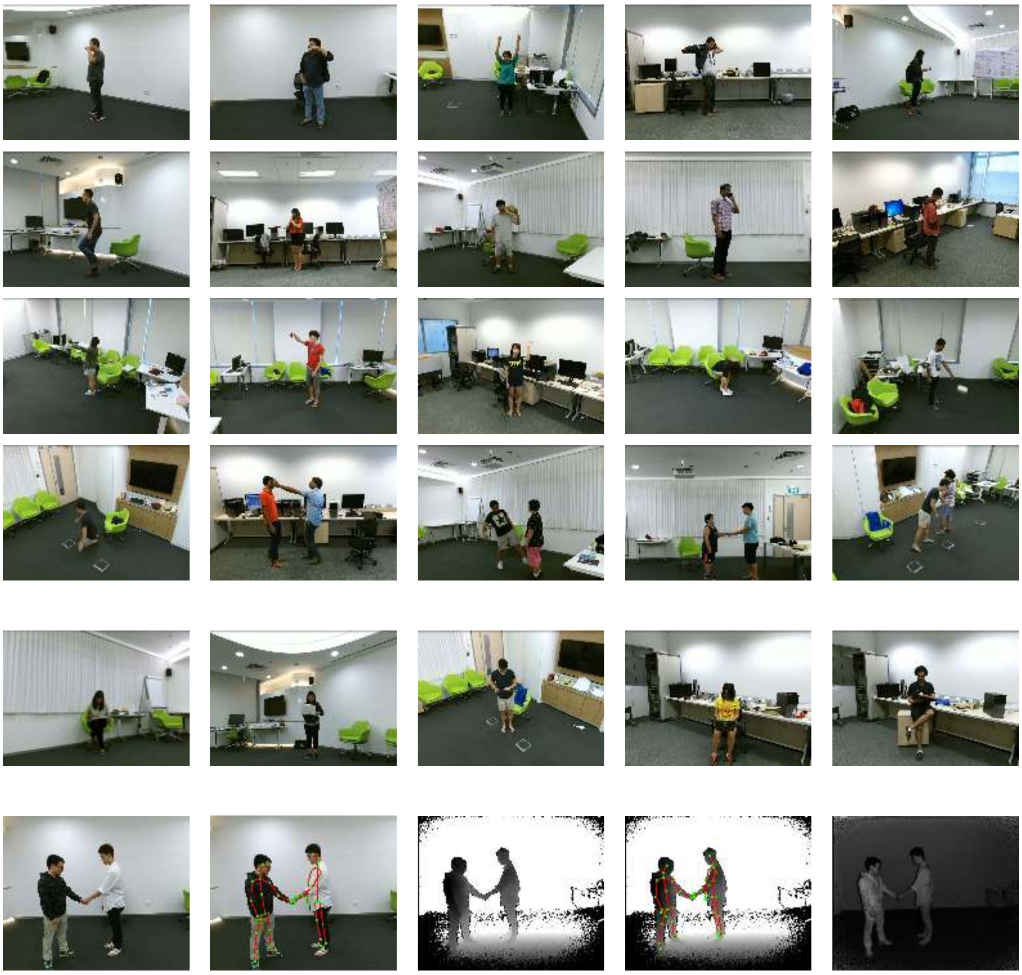


Fig. 5.5 Sample RGB frames from our dataset. First four rows show the variety in human subjects and camera views. Fifth row depicts the intra-class variation of the performances. The last row illustrates RGB, RGB+joints, depth, depth+joints, and IR modalities of a sample frame.

Although the proposed dataset mitigates a lot of limitations in depth-based action recognition benchmark datasets, it still has a number of limitations. First is the artificial nature of the performed actions. All the action samples are performed by actors as subjects. The ideal collection is supposed to be done in a free-running situation to capture real actions when subjects are performing them. Next, in our dataset, we limited our action classes to the ones with vertical body poses to help the Kinect's SDK to track and extract more precise skeleton data. That is why we did not have action classes like "going to bed", "wake up", "crawling", etc. in our dataset. The same reason prevented us from including top and back views of the actions in our dataset. Finally, lack of an efficient video compression for depth sequences made us to store the depth-maps frame by frame which leads to a very large file sizes and makes it troublesome to transfer the dataset over the web. An ideal depth sequence video compression technique can help to reduce the size of the dataset drastically.

The work proposed in this chapter is published in [81].

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this thesis, we proposed various methods to improve human activity analysis in depth-based videos. The major contributions of this thesis are focused on feature analysis in different depth-based and RGB+D based scenarios for action recognition.

First, we introduced an integrated learning and classification framework on multiple depth-based input features of human actions. Our proposed method utilized hierarchical mixed-norm to perform multimodal multipart learning over a set of extracted features. The intuition of part-sparsity was utilized in the higher layer of the proposed hierarchical mixed-norm to apply sparsity between the features of various body parts to perform a soft part selection. Within each group of part-based features, a two-level mixed norm was used to apply a diversity norm on the elements of each input feature type while regularizing the weight decay of their combination by the middle level norm. The proposed learning framework integrates the group feature selection and weight regularization within the classifier learning. The reported results on three benchmark datasets, showed the effectiveness of our proposed learning framework in applying soft part selection and utilizing the prior structure of the input features into the learning step.

Second, we studied the mutual properties of depth-based features with RGB-based counterparts for the task of action recognition. We proposed a new deep learning framework to perform a hierarchical correlated-independent component factorization, in order to achieve a better representation of input RGB+D features. In each layer, the proposed network factorized the input multimodal features into their correlated and independent components in an autoencoder based analysis framework. An advantage of the proposed factorization technique is its ability to be applied in a convolutional setting over temporal and spatial axes. To utilize the independence of derived hierarchical components in the classification step, we proposed a structured sparsity based classifier. Our suggested classifier regularized the learning by applying mixed-norms to perform soft component and layer-wise group selection to achieve optimal fusion of factorized components. We evaluated our method on four RGB+D action recognition benchmark datasets, in which we showed the strength of the proposed deep correlation-independence network and the suggested structured sparsity learning machine. We achieved the state-of-the-art performances on all the evaluated benchmarks.

Third, to alleviate the size-related limitations of depth-based human activity analysis datasets for applying data-driven learning techniques, we proposed the first large-scale RGB+D action recognition dataset . This dataset includes more than 50K human video samples and more than 4 million video frames of human activities. We invited 40 distinct human subjects to perform 60 different action classes. The samples of this dataset are highly variant in the quality of action performance and camera views. To the best of our knowledge, ours is the largest RGB+D human activity analysis dataset in all of the comparison aspects. The scales of the collected dataset makes it possible to apply data-hungry analysis methods *e.g.* deep learning networks on this problem and surpass hand-crafted feature representations in performance accuracies.

Lastly, we proposed a new recurrent network based learning framework for skeleton-based human action recognition. Based on the intuition of the high correlation among the motion of the body joints within each limb, and low correlation between the limbs, our part-aware LSTM applies the grouping of input body configuration inside LSTM units to improve the performance of the action learning framework. Our experimental evaluations of this method on our proposed dataset, first showed the effectiveness of the large-scale data in applying data-driven learning frameworks, and second, showed the strength of the proposed P-LSTM model over traditional recurrent models.

6.2 Future Work

In this section, we introduce some of our potential research directions to follow in future.

6.2.1 Convolutional Networks for Depth Videos

The introduction of our large scale RGB+D action dataset, removes the barriers of applying various deep learning techniques. The most successful classes of deep networks in visual recognition tasks are convolutional networks [27].

Although some attempts have been done so far, but the small scales of depth based action datasets prevented us to apply deep convnets on these type of data directly. One of our directions in future is to apply convnets for depth based action recognition using our dataset.

Multitask learning frameworks can also be applied in a similar fashion to utilize the trained deep networks on the larger scale dataset for learning features and classifiers on smaller sized datasets jointly.

6.2.2 Learning 3D Skeleton Configurations for Better Action Recognition

The collected dataset provides three different horizontal views of the same action at the same time. In the current evaluation setup of the dataset this correspondence is ignored and the three views are considered as different samples.

Utilizing this information can help us to improve the 3D skeletons to more accurate annotations as the precise ground truth body pose representation. Applying different data-driven techniques (*e.g.* convnets, recurrent nets, and their combination) we can train a network to extract more reliable body poses from a sequence of depth maps in videos. The trained network can be effectively used to extract better body pose configurations on other datasets which can lead to better performance in skeleton-based action recognition.

This idea can easily be extended into skeleton learning from RGB+D by learning skeleton on jointly on depth and RGB videos.

6.2.3 Joint-based Recurrent Networks

The proposed framework in section 5.3 can be further improved by learning recurrent cells for each joint. In a deep joint-based recurrent network, each cell will receive the hidden state of the previous joint in a predefined sequence of body joints, as well as the hidden states of the same cell in previous time step, and on previous network layer, and learns how to project its input to the optimal hidden representation at each step.

In this fashion, each recurrent unit will learn the spatial and temporal behavior of the corresponding body joint and its relation with other joints in order to maximize the discriminative strength of the final layer's representation.

This idea is developed during the examination period of this thesis and published in [55, 54].

Author's Publications

- **Amir Shahroudy**, Tian-Tsong Ng, Qingxiong Yang, and Gang Wang, "Multimodal Multipart Learning for Action Recognition in Depth Videos", *to appear in IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI).*

- **Amir Shahroudy**, Jun Liu, Tian-Tsong Ng, and Gang Wang, "NTU RGB+D: A Large Scale Dataset for 3D Human Activity Analysis", *in IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, Las Vegas, US.

- **Amir Shahroudy**, Tian-Tsong Ng, Yihong Gong, and Gang Wang, "Deep Multimodal Feature Analysis for Action Recognition in RGB+ D Videos", *under revision in IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI).*

- **Amir Shahroudy**, Gang Wang, and Tian-Tsong Ng, "Multi-Modal Feature Fusion for Action Recognition in RGB-D Sequences", *in 6th International Symposium on Communications, Control and Signal Processing (ISCCSP14)*, 2014, Athens, Greece.

- Jun Liu, **Amir Shahroudy**, Dong Xu, Gang Wang, "Spatio-Temporal LSTM with Trust Gates for 3D Human Action Recognition", *accepted in European Conference on Computer Vision (ECCV), 2016, Amsterdam, Netherlands.*

- Jun Liu, **Amir Shahroudy**, Dong Xu, Gang Wang, "Skeleton-Based Action Recognition Using Spatio-Temporal LSTM Network with Trust Gates", *Submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI).*

References

[1] A. Abdulnabi, G. Wang, J. Lu, and K. Jia. Multi-task cnn model for attribute prediction. *IEEE Transactions on Multimedia (TMM)*, 2015.

[2] J. Aggarwal and L. Xia. Human activity recognition from 3d data: A review. *Pattern Recognition Letters*, 48:70–80, 2014.

[3] S. Althloothi, M. H. Mahoor, X. Zhang, and R. M. Voyles. Human activity recognition using multi-features and multiple kernel learning. *Pattern Recognition*, 2014.

[4] G. Andrew, R. Arora, J. Bilmes, and K. Livescu. Deep canonical correlation analysis. In *International Conference on Machine Learning (ICML)*, 2013.

[5] F. R. Bach and M. I. Jordan. A probabilistic interpretation of canonical correlation analysis. 2005.

[6] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle, et al. Greedy layer-wise training of deep networks. *Advances in neural information processing systems (NIPS)*, 2007.

[7] V. Bloom, V. Argyriou, and D. Makris. *G3Di: A Gaming Interaction Dataset with a Real Time Detection and Evaluation Framework*. 2015.

[8] A. Bobick and J. Davis. The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2001.

[9] M. Borga. Canonical correlation: a tutorial. *Online tutorial*, 2001.

[10] A. Bosch, A. Zisserman, and X. Munoz. Representing shape with a spatial pyramid kernel. In *ACM International Conference on Image and Video Retrieval (CIVR)*, 2007.

[11] W. Byeon, T. M. Breuel, F. Raue, and M. Liwicki. Scene labeling with lstm recurrent neural networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[12] Z. Cai, J. Han, L. Liu, and L. Shao. Rgb-d datasets using microsoft kinect or similar sensors: a survey. *Multimedia Tools and Applications*, pages 1–43, 2016.

[13] Z. Cai, L. Wang, X. Peng, and Y. Qiao. Multi-view super vector for action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[14] R. Caruana. Multitask learning. *Machine Learning*, 1997.

[15] S. Chatzis. Infinite markov-switching maximum entropy discrimination machines. In *International Conference on Machine Learning (ICML)*, 2013.

[16] C. Chen, R. Jafari, and N. Kehtarnavaz. Utd-mhad: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor. In *IEEE*

International Conference on Image Processing (ICIP), Sept 2015.

[17] L. Chen, H. Wei, and J. Ferryman. A survey of human motion analysis using depth imagery. *Pattern Recognition Letters*, 2013.

[18] Z. Cheng, L. Qin, Y. Ye, Q. Huang, and Q. Tian. Human daily action analysis with multi-view and color-depth data. In A. Fusiello, V. Murino, and R. Cucchiara, editors, *European Conference on Computer Vision Workshops and Demonstrations (ECCV Workshops)*, volume 7584 of *Lecture Notes in Computer Science*, pages 52–61. Springer Berlin Heidelberg, 2012.

[19] R. Collobert, K. Kavukcuoglu, and C. Farabet. Torch7: A matlab-like environment for machine learning. In *BigLearn, Advances in Neural Information Processing Systems Workshop (NIPS Workshop)*, 2011.

[20] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[21] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[22] Y. Du, W. Wang, and L. Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1110–1118, June 2015.

[23] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.

[24] G. Evangelidis, G. Singh, and R. Horaud. Skeletal quads: Human action recognition using joint quadruples. In *International Conference on Pattern Recognition (ICPR)*, pages 4513–4518, Aug 2014.

[25] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research (JMLR)*, 9:1871–1874, 2008.

[26] S. Gao, I. Tsang, L.-T. Chia, and P. Zhao. Local features are not lonely - laplacian sparse coding for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[27] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, and G. Wang. Recent Advances in Convolutional Neural Networks. *ArXiv e-prints*, 2015.

[28] F. Han, B. Reily, W. Hoff, and H. Zhang. Space-Time Representation of People Based on 3D Skeletal Data: A Review. *arXiv e-prints*, 2016.

[29] J. Han, L. Shao, D. Xu, and J. Shotton. Enhanced computer vision with microsoft kinect sensor: A review. *IEEE Transactions on Cybernetics*, 2013.

[30] D. Hardoon, S. Szedmak, and J. Shawe-Taylor. Canonical correlation analysis: An overview with application to learning methods. *Neural Computation*, 2004.

[31] G. Hinton, S. Osindero, and Y.-W. Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 2006.

[32] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, Nov 1997.

[33] H. Hotelling. Relations between two sets of variates. *Biometrika*, 1936.

[34] J.-F. Hu, W.-S. Zheng, J. Lai, and J. Zhang. Jointly learning heterogeneous features for rgb-d activity recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[35] Y. Jia, M. Salzmann, and T. Darrell. Factorized latent spaces with structured sparsity. In *Advances in Neural Information Processing Systems (NIPS)*, 2010.

[36] A. Karpathy, J. Johnson, and F. Li. Visualizing and understanding recurrent networks. *arXiv*, abs/1506.02078, 2015.

[37] A. Klaeser, M. Marszalek, and C. Schmid. A spatio-temporal descriptor based on 3d-gradients. In *British Machine Vision Conference (BMVC)*, 2008.

[38] Y. Kong and Y. Fu. Bilinear heterogeneous information machine for rgb-d action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[39] H. S. Koppula, R. Gupta, and A. Saxena. Learning human activities and object affordances from rgb-d videos. *The International Journal of Robotics Research (IJRR)*, 32(8):951–970, 2013.

[40] D. Kosmopoulos, P. Doliotis, V. Athitsos, and I. Maglogiannis. Fusion of color and depth video for human behavior recognition in an assistive environment. In *Distributed, Ambient, and Pervasive Interactions*, Lecture Notes in Computer Science. 2013.

[41] M. Kowalski. Sparse regression using mixed norms. *Applied and Computational Harmonic Analysis*, 2009.

[42] M. Kowalski and B. Torrésani. Structured Sparsity: from Mixed Norms to Structured Shrinkage. In *Signal Processing with Adaptive Sparse Structured Representations (SPARS)*, 2009.

[43] M. Kowalski and B. Torrésani. Sparsity and persistence: mixed norms provide simple signal models with dependent coefficients. *Signal, Image and Video Processing*, 2009.

[44] P. L. Lai and C. Fyfe. Kernel and nonlinear canonical correlation analysis. *International Journal of Neural Systems*, 2000.

[45] I. Laptev and T. Lindeberg. Space-time interest points. In *IEEE International Conference on Computer Vision (ICCV)*, 2003.

[46] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

[47] Q. Le, W. Zou, S. Yeung, and A. Ng. Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[48] Q. V. Le, A. Karpenko, J. Ngiam, and A. Y. Ng. Ica with reconstruction cost for efficient overcomplete feature learning. In *Advances in Neural Information Processing Systems (NIPS)*. 2011.

[49] H. Lee, C. Ekanadham, and A. Y. Ng. Sparse deep belief net model for visual area v2. In *Advances in Neural Information Processing Systems (NIPS)*. 2008.

[50] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *International Conference on Machine Learning (ICML)*, 2009.

[51] W. Li, Z. Zhang, and Z. Liu. Action recognition based on a bag of 3d points. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops)*, 2010.

[52] H. Liu, M. Palatucci, and J. Zhang. Blockwise coordinate descent procedures for the multi-task lasso, with applications to neural semantic basis discovery. In *International Conference on Machine Learning (ICML)*, 2009.

[53] H. Liu, M. Yuan, and F. Sun. Rgb-d action recognition using linear coding. *Neurocomputing*, 2015.

[54] J. Liu, A. Shahroudy, D. Xu, and G. Wang. Skeleton-based action recognition using spatio-temporal lstm network with trust gates.

[55] J. Liu, A. Shahroudy, D. Xu, and G. Wang. Spatio-temporal lstm with trust gates for 3d human action recognition. In *European Conference on Computer Vision (ECCV)*. 2016.

[56] L. Liu and L. Shao. Learning discriminative representations from rgb-d video data. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence (IJCAI)*, 2013.

[57] T. Liu, R. R. Varior, and G. Wang. Visual tracking using learned color features. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[58] C. Lu, J. Jia, and C.-K. Tang. Range-sample depth feature for action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[59] R. Lun and W. Zhao. A survey of applications and human motion recognition with microsoft kinect. *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)*, 2015.

[60] J. Luo, W. Wang, and H. Qi. Group sparsity and geometry constrained dictionary learning for action recognition from depth maps. In *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[61] M. Meng, H. Drira, M. Daoudi, and J. Boonaert. Human-object interaction recognition by learning the distances between the object and the skeleton joints. In *IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, volume 07, pages 1–6, May 2015.

[62] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng. Multimodal deep learning. In *International Conference on Machine Learning (ICML)*, 2011.

[63] B. Ni, G. Wang, and P. Moulin. Rgbd-hudaact: A color-depth video database for human daily activity recognition. In *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2011.

[64] G. Obozinski, B. Taskar, and M. I. Jordan. Joint covariate selection and joint subspace selection for multiple classification problems. *Statistics and Computing*, 2010.

[65] E. Ohn-Bar and M. Trivedi. Joint angles similarities and hog^2 for action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops)*, 2013.

[66] A. V. Oppenheim, R. W. Schafer, and J. R. Buck. *Discrete-time Signal Processing (2Nd Ed.)*. Prentice-Hall, Inc., 1999.

[67] O. Oreifej and Z. Liu. Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[68] X. Peng, L. Wang, X. Wang, and Y. Qiao. Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice. *arXiv*, abs/1405.4506, 2014.

[69] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[70] H. Rahmani, A. Mahmood, D. Huynh, and A. Mian. Action classification with locality-constrained linear coding. In *International Conference on Pattern Recognition (ICPR)*, 2014.

[71] H. Rahmani, A. Mahmood, D. Huynh, and A. Mian. Histogram of oriented principal components for cross-view action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2016.

[72] H. Rahmani, A. Mahmood, D. Q. Huynh, and A. Mian. Real time action recognition using histograms of depth gradients and random decision forests. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2014.

[73] H. Rahmani, A. Mahmood, D Q Huynh, and A. Mian. Hopc: Histogram of oriented principal components of 3d pointclouds for action recognition. In *European Conference on Computer Vision (ECCV)*. 2014.

[74] H. Rahmani and A. Mian. Learning a non-linear knowledge transfer model for cross-view action recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[75] H. Rahmani and A. Mian. 3d action recognition from novel viewpoints. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[76] A. Rakotomamonjy, R. Flamary, G. Gasso, and S. Canu. $l_p - l_q$ penalty for sparse linear and sparse multiple kernel multitask learning. *IEEE Transactions on Neural Networks*,

2011.

[77] M. Ranzato, C. Poultney, S. Chopra, and Y. LeCun. Efficient learning of sparse representations with an energy-based model. In *Advances in Neural Information Processing Systems (NIPS)*, 2007.

[78] B. Rao and K. Kreutz-Delgado. An affine scaling methodology for best basis selection. *IEEE Transactions on Signal Processing (TSP)*, 1999.

[79] M. Salzmann, C. H. Ek, R. Urtasun, and T. Darrell. Factorized orthogonal latent spaces. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010.

[80] M. Schmidt. Minfunc, 2005.

[81] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang. Ntu rgb+d: A large scale dataset for 3d human activity analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[82] A. Shahroudy, T.-T. Ng, Y. Gong, and G. Wang. Deep Multimodal Feature Analysis for Action Recognition in RGB+D Videos. *arXiv e-prints*, 2016.

[83] A. Shahroudy, T. T. Ng, Q. Yang, and G. Wang. Multimodal multipart learning for action recognition in depth videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, PP(99):1–1, 2016.

[84] A. Shahroudy, G. Wang, and T.-T. Ng. Multi-modal feature fusion for action recognition in rgb-d sequences. In *International Symposium on Communications, Control and Signal Processing (ISCCSP)*, 2014.

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

[86] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems (NIPS)*. 2014.

[87] J. Sivic and A. Zisserman. Video google: a text retrieval approach to object matching in videos. In *IEEE International Conference on Computer Vision (ICCV)*, 2003.

[88] Y. Song, S. Liu, and J. Tang. Describing trajectory of surface patch for human action recognition on rgb and depth videos. *IEEE Signal Processing Letters (SPL)*, 2015.

[89] N. Srivastava and R. Salakhutdinov. Multimodal learning with deep boltzmann machines. *Journal of Machine Learning Research (JMLR)*, 2014.

[90] J. Sung, C. Ponce, B. Selman, and A. Saxena. Human activity detection from rgbd images. In *AAAI Conference on Artificial Intelligence Workshops (AAAI Workshops)*, 2011.

[91] I. Sutskever, O. Vinyals, and Q. V. V. Le. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, editors, *Advances in Neural Information Processing Systems (NIPS)*, pages 3104–3112. Curran Associates, Inc., 2014.

[92] M. Szafranski, Y. Grandvalet, and P. Morizet-mahoudeaux. Hierarchical penalization. In *Advances in Neural Information Processing Systems (NIPS)*. 2008.

[93] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 1996.

[94] J.-S. Tsai, Y.-P. Hsu, C. Liu, and L.-C. Fu. An efficient part-based approach to action recognition from rgb-d video with bow-pyramid representation. In *International Conference on Intelligent Robots and Systems (IROS)*, 2013.

[95] R. R. Varior, M. Haloi, and G. Wang. Gated siamese convolutional neural network architecture for human re-identification. In *European Conference on Computer Vision (ECCV)*, 2016.

[96] R. R. Varior, B. Shuai, J. Lu, D. Xu, and G. Wang. A siamese long short-term memory architecture for human re-identification. In *European Conference on Computer Vision (ECCV)*, 2016.

[97] R. R. Varior and G. Wang. A data-driven color feature learning scheme for image retrieval. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[98] R. R. Varior, G. Wang, J. Lu, and T. Liu. Learning invariant color features for person reidentification. *IEEE Transactions on Image Processing (TIP)*, 2016.

[99] V. Veeriah, N. Zhuang, and G.-J. Qi. Differential recurrent neural networks for action recognition. In *The IEEE International Conference on Computer Vision (ICCV)*, 2015.

[100] R. Vemulapalli, F. Arrate, and R. Chellappa. Human action recognition by representing 3d skeletons as points in a lie group. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[101] C. Wang, Y. Wang, and A. L. Yuille. Mining 3d key-pose-motifs for action recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[102] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu. Dense trajectories and motion boundary descriptors for action recognition. *International Journal of Computer Vision (IJCV)*, 2013.

[103] H. Wang, F. Nie, W. Cai, and H. Huang. Semi-supervised robust dictionary learning via efficient $l_{2,0^+}$-norms minimization. In *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[104] H. Wang, F. Nie, and H. Huang. Multi-view clustering and feature learning via structured sparsity. In *International Conference on Machine Learning (ICML)*, 2013.

[105] H. Wang, F. Nie, and H. Huang. Robust and discriminative self-taught learning. In *International Conference on Machine Learning (ICML)*, 2013.

[106] H. Wang, F. Nie, H. Huang, and C. Ding. Heterogeneous visual features fusion via sparse multimodal machine. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[107] H. Wang and C. Schmid. Action recognition with improved trajectories. In *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[108] J. Wang, Z. Liu, J. Chorowski, Z. Chen, and Y. Wu. Robust 3d action recognition with random occupancy patterns. In *European Conference on Computer Vision (ECCV)*, 2012.

[109] J. Wang, Z. Liu, Y. Wu, and J. Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[110] J. Wang, Z. Liu, Y. Wu, and J. Yuan. Learning actionlet ensemble for 3d human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2014.

[111] J. Wang, X. Nie, Y. Xia, Y. Wu, and S.-C. Zhu. Cross-view action modeling, learning, and recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2649–2656, June 2014.

[112] J. Wang and Y. Wu. Learning maximum margin temporal warping for action recognition. In *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[113] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[114] K. Wang, X. Wang, L. Lin, M. Wang, and W. Zuo. 3d human activity recognition with reconfigurable convolutional neural networks. In *ACM International Conference on Multimedia (ACM MM)*, 2014.

[115] K. Wang, X. Wang, L. Lin, M. Wang, and W. Zuo. 3d human activity recognition with reconfigurable convolutional neural networks. In *Proceedings of the ACM International Conference on Multimedia*, MM '14, pages 97–106, New York, NY, USA, 2014. ACM.

[116] L. Wang, N. T. Pham, T.-T. Ng, G. Wang, K. L. Chan, and K. Leman. Learning deep features for multiple object tracking by using a multi-task learning strategy. In *IEEE International Conference on Image Processing (ICIP)*, 2014.

[117] P. Wang, W. Li, Z. Gao, J. Zhang, C. Tang, and P. Ogunbona. Action recognition from depth maps using deep convolutional neural networks. In *IEEE Transactions on Human Machine Systems (THMS)*, 2015.

[118] P. Wei, Y. Zhao, N. Zheng, and S.-C. Zhu. Modeling 4d human-object interactions for event and object recognition. In *IEEE International Conference on Computer Vision (ICCV)*, pages 3272–3279, Dec 2013.

[119] C. Wolf, E. Lombardi, J. Mille, O. Celiktutan, M. Jiu, E. Dogan, G. Eren, M. Baccouche, E. Dellandréa, C.-E. Bichot, C. Garcia, and B. Sankur. Evaluation of video activity localizations integrating quality and quantity measurements. *Computer Vision and Image Understanding (CVIU)*, 2014.

[120] L. Xia and J. Aggarwal. Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[121] L. Xia, C.-C. Chen, and J. Aggarwal. View invariant human action recognition using histograms of 3d joints. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops)*, 2012.

[122] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[123] X. Yang and Y. Tian. Eigenjoints-based action recognition using naïve-bayes-nearest-neighbor. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops)*, 2012.

[124] X. Yang and Y. Tian. Effective 3d action recognition using eigenjoints. *Journal of Visual Communication and Image Representation*, 2014.

[125] X. Yang and Y. Tian. Super normal vector for activity recognition using depth sequences. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[126] X. Yang, C. Zhang, and Y. Tian. Recognizing actions using depth motion maps-based histograms of oriented gradients. In *ACM International Conference on Multimedia (MM)*, 2012.

[127] M. Ye, Q. Zhang, L. Wang, J. Zhu, R. Yang, and J. Gall. A survey on human motion analysis from depth data. In *Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications*. 2013.

[128] G. Yu, Z. Liu, and J. Yuan. Discriminative orderlet mining for real-time recognition of human-object interaction. In *Asian Conference on Computer Vision (ACCV)*, 2014.

[129] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2006.

[130] X.-T. Yuan, X. Liu, and S. Yan. Visual classification with multitask joint sparse representation. *IEEE Transactions on Image Processing (TIP)*, 2012.

[131] K. Yun, J. Honorio, D. Chattopadhyay, T. L. Berg, and D. Samaras. Two-person interaction detection using body-pose features and multiple instance learning. In *The IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops)*, 2012.

[132] J. Zhang, W. Li, P. O. Ogunbona, P. Wang, and C. Tang. Rgb-d-based action recognition datasets: A survey. *arXiv*, abs/1601.05511, 2016.

[133] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja. Robust visual tracking via multi-task sparse learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[134] Z. Zhang. Microsoft kinect sensor and its effect. *IEEE MultiMedia*, 2012.

[135] P. Zhao, G. Rocha, and B. Yu. The composite absolute penalties family for grouped and hierarchical variable selection. *The Annals of Statistics*, 2009.

[136] Y. Zhao, Z. Liu, L. Yang, and H. Cheng. Combing rgb and depth map features for human activity recognition. In *Signal Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2012.

[137] Z. Y. Zhao Runlin. Depth induced feature representation for 4d human activity recognition. *Computer Modelling & New Technologies*, 2014.

[138] Q. Zhou, G. Wang, K. Jia, and Q. Zhao. Learning to share latent tasks for action recognition. In *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[139] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie. Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks. *AAAI Conference on Artificial Intelligence*, 2016.

[140] Y. Zhu, W. Chen, and G. Guo. Fusing multiple features for depth-based action recognition. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2015.

[141] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2005.