

POLYDOMAIN LEARNING FOR HUMAN ACTIVITY RECOGNITION

Thesis Submitted for the Award of the Degree of

DOCTOR OF PHILOSOPHY

in

Electronics and Communication Engineering

By

Rosepreet Kaur Bhogal

Registration Number: 41700216

Supervised By

Dr. V Devendran

Professor

Lovely Professional University, India



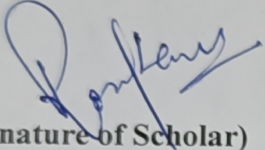
L OVELY
P ROFESSIONAL
U NIVERSITY

Transforming Education Transforming India

**LOVELY PROFESSIONAL UNIVERSITY, PUNJAB
2023**

DECLARATION

I, hereby declared that the presented work in the thesis entitled “**POLYDOMAIN LEARNING FOR HUMAN ACTIVITY RECOGNITION**” in fulfilment of degree of **Doctor of Philosophy (Ph. D.)** is outcome of research work carried out under the supervision **Dr V Devendran**, working as Professor, in the School of Computer Science and Engineering of Lovely Professional University, Punjab, India. In keeping with general practice of reporting scientific observations, due acknowledgements have been made whenever work described here has been based on findings of other investigator. This work has not been submitted in part or full to any other University or Institute for the award of any degree.



(Signature of Scholar)

Name of the scholar: Rosepreet Kaur Bhogal

Registration No.: 41700216

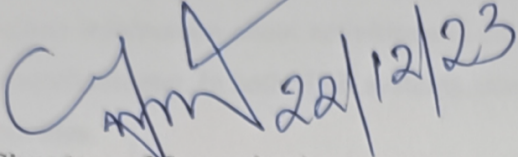
Department/school: School of Electronics and Electrical Engineering

Lovely Professional University

Punjab, India

CERTIFICATE

This is to certify that the work reported in the Ph. D. thesis entitled “**POLYDOMAIN LEARNING FOR HUMAN ACTIVITY RECOGNITION**” submitted in fulfillment of the requirement for the reward of degree of **Doctor of Philosophy (Ph.D.)** in the School of Electronics and Electrical Engineering, is a research work carried out by **Rosepreet Kaur Bhogal, 41700216**, is a bonafide record of his/her original work carried out under my supervision and that no part of thesis has been submitted for any other degree, diploma or equivalent course.



(Signature of Supervisor)

Name of supervisor: Dr V Devendran

Designation: Professor

Department/school: School of Computer Science and Engineering

University: Lovely Professional University, Phagwara, India

ABSTRACT

The thesis entitled “Polydomain Learning for Human Activity Recognition” is a model that can recognise human activities from many views and modalities. Now a day, Surveillance has become an essential need for smart cities. It is required to improve life and live life in a safe zone. Humans engage in various activities that can be normal as well as abnormal. The video-driven system can help in healthcare, surveillance, transportation, factory, schools, malls, marts, and any human-computer interaction. There are mainly two categories of human action recognition (HAR) systems based on equipment. First, vision-based HAR is already available in various places to capture video and use it for surveillance purposes. And second is a sensor-based HAR system which is through smartphones. Built-in smartphone sensors capture information about activities and can retrieve from the smartphone's inbuilt gyroscope or accelerometer. In both HAR systems, machine learning algorithms can recognize actions or activities.

To begin, we analyse, judge and contrast diverse cutting-edge procedures relying on numerical assessment and video datasets. The most popular approach is a convolutional neural network (CNN) and Long short-term memory (LSTM). Nevertheless, if the setup is crafted on a deep learning model. Then, there is a requirement for big data. That is why this work carried out with the NTURGB+D dataset. The dataset contains four modalities and three views comprising a polydomain learning system. The other dataset also has been used for activity recognition and object detection i.e., HMDB51 and CamVid, respectively.

Next secondly, we propose a subject segmentation method for RGB video frames using skeletal 3D detail of NTURGB+D. In this study, we designed an automated system for action segmentation from videos. The window size is flexible and determined by the video. The NTURGB+D dataset was utilized to present the experimental data. We conducted subject segmentation using the 3D skeletal information from NTURGB+D RGB videos. The results of the experiments showed the performance and evaluation of 5 randomly selected action videos from the dataset.

The approach of object detection based on semantic segmentation is assessed by using a DeepLab v3+ network with Resnet18 for weight initialization. To detect various scenes from

video frames, the CamVid database is employed. This database contains 701 RGB frames with labeled RGB values and pixel-wise segmentation tags. To evaluate the model's effectiveness, three metrics were considered: accuracy, IoU, and score.

Fourthly, the analysis performs for object detection using features i.e., HOF (Histogram of optical flow). This work analyses four motion estimating optical flow methods (Farneback, Horn Schunck, Lucas Kanade, and Lucas-Kanade Derivative of Gaussian) explored based on visualization and PSNR. The analysis of experimental results was conducted using the NTURGB+D dataset. Finally, an exploration of HOG (histogram of oriented gradient) as a method for object detection in video frames was undertaken.

Fifth, we propose a model that can recognize actions by transfer learning. The architecture was designed using BiLSTM layers, which help in learning the system based on time dependencies. A pre-trained Google Net network was used to transform each frame into a tagged vector. The HMDB51 dataset was used to conduct an evaluation of the model. This analysis yielded an accuracy of 93.04% for the 10 classes and 63.96% for the 51 classes in the dataset.

Sixth, we propose using a transversal tree from 3D skeleton data to represent videos in a sequence. The authors proposed two neural network architectures: CNN_RNN_1, which is utilized to identify the optimal features, and CNN_RNN_2, which is employed to classify actions. The proposed deep neural network-based model, CNN_RNN_1 and CNN_RNN_2, utilizes a convolutional neural network (CNN), Long short-term memory (LSTM), and Bidirectional Long short-term memory (BiLSTM) layered structure to successfully obtain the desired accuracy of 88.89%. An analysis of the performance of the proposed model was conducted against existing state-of-the-art models using the NTURGB+D dataset. This dataset is one of the prominent ones used to recognize human activities. Furthermore, the comparison results displayed that the proposed model was more effective than the current leading models.

Last, the two-stream network for polydomain learning has been proposed for action recognition. The system can take input from two modalities, i.e., RGB videos and Skeleton 3D data. The proposed models in chapters 5 and 6 have been used to design the two-stream network for polydomain learning. The recognition rate is obtained after the model evaluation is 85.76%. A detailed description of the model is given in chapter 6, section 6.7.

ACKNOWLEDGEMENT

Firstly, I start in the name of the Almighty God, the most Gracious, the most Merciful, for bestowing upon me good health and mental state to carry on this research.

I would like to express a sincere acknowledgement to my supervisor, Dr. V. Devendran, Professor, School of computer science and engineering because he gave me the opportunity to research under his guidance and supervision. I received motivation encouragement and support from him during all my studies. With him, I have learned writing papers for Journal/conferences and sharing my ideas to the public. I insist on thanking them for their understanding and support throughout the research process. They have always been by my side whenever I needed them, providing me with their full support, ideas and suggestions to improve my work. They have always encouraged me to participate in the relevant international conferences and courses that improved my knowledge, skills and expertise in my domain and over certain tools. Without their continuous support, guidance and help I would have never been able to achieve the reward of completing my Ph.D.

I would like to thank my husband, Mr. Ajmer Singh who has given unconditional support and motivation. Without him, I could not have been focused and achieved this. I am pleased to thank my colleagues and head, which helped me directly or indirectly during this journey.

During the Ph.D. studies in the Lovely Professional University of several persons and institutions collaborated directly and indirectly with my research. Without their support it would be impossible for me to finish my work. That is why I wish to dedicate this section to recognize their support.

At last, but the most important I would like to thank my mother, brothers, Father in law, mother in law and whole family for their unconditional support, inspiration and love.



Rosepreet Kaur Bhogal

CONTENTS

| Table of content | Page No. |
|---|-----------------|
| Declaration | ii |
| Certificate | iii |
| Abstract | iv |
| Acknowledgment | vi |
| List of Tables | x |
| List of Figures | xi |
| 1 Introduction | 1 |
| 1.1 Problem Statement | 1 |
| 1.2 Notion of Human Actions and Activities | 1 |
| 1.3 Applications of activity recognition | 2 |
| 1.4 Notion of Activity Recognition System | 4 |
| 1.5 Deep learning | 5 |
| 1.5.1 Polydomain learning (PDL) | 6 |
| 1.6 Data modalities | 7 |
| 1.7 Notion of research gap | 7 |
| 1.8 Thesis contribution | 8 |
| 1.9 Objectives of the study | 8 |
| 1.10 Thesis structure | 9 |
| 2 Literature Survey | 11 |
| 2.1 Survey of HAR System | 11 |
| 2.2 Research Gap | 23 |
| 3 Subject segmentation of actions from RGB frames with skeleton data | 24 |
| 3.1 Subject Segmentation of RGB frames | 24 |
| 3.1.1 Literature Survey | 24 |
| 3.1.2 About Dataset NTURGB+D | 25 |
| 3.1.3 Steps for action segmentation | 26 |
| 3.1.4 Experimental Analysis | 26 |
| 3.2 Semantic Segmentation for Object Recognition | 30 |
| 3.2.1 Literature Survey | 30 |

| | | |
|----------|--|----|
| 3.2.2 | Steps for Object recognition using semantic segmentation | 32 |
| 3.2.3 | About Dataset CamVid | 33 |
| 3.2.4 | Experimental Analysis | 34 |
| 3.3 | Summary | 39 |
| 4 | Feature Extraction for object detection | 41 |
| 4.1 | HOF (Histogram optical flow) features | 41 |
| 4.1.1 | Methods for HOF | 42 |
| 4.1.2 | Experimental Analysis | 44 |
| 4.2 | HOG (Histogram of oriented gradients) features | 50 |
| 4.3 | Summary | 50 |
| 5 | Human Activity Recognition using BiLSTM Network | 52 |
| 5.1 | Literature Survey | 53 |
| 5.2 | Feature Extraction for HAR | 53 |
| 5.3 | BiLSTM Network for recognition using transfer learning | 54 |
| 5.4 | Experimental Analysis | 56 |
| 5.4.1 | Implementation details | 56 |
| 5.4.2 | About dataset HMDB51 | 57 |
| 5.4.3 | Conduct an analysis for 10 classes of HMDB51 | 58 |
| 5.4.4 | Conduct an analysis for 51 classes of HMDB51 | 59 |
| 5.4.5 | Comparison with other state of art methods | 60 |
| 5.5 | Summary | 60 |
| 6 | Action Recognition for Multiview using NTURGB+D Dataset | 61 |
| 6.1 | Literature Survey | 61 |
| 6.2 | Proposed research methodology | 64 |
| 6.2.1 | 3D Skeleton Pre-processing (Representation of 3D Skeleton Data into Sequences Using Transversal Tree) | 66 |
| 6.2.2 | CNN_RNN_1 for Optimal Features | 66 |
| 6.2.3 | CNN_RNN_2 for Action Classification | 67 |
| 6.3 | Experiment Analysis | 68 |
| 6.3.1 | About Dataset NTURGB+D | 68 |
| 6.3.2 | Implementation Details | 69 |
| 6.3.3 | Results | 69 |
| 6.4 | Comparing our approach to other existing methods | 71 |

| | |
|--|-----------|
| 6.5 Activity ID with Activity name in dataset (NTURGB+D) | 71 |
| 6.6 Confusion matrix | 72 |
| 6.7 Two stream model for Polydomain learning | 74 |
| 6.7.1 Experimental analysis | 75 |
| 6.8 Summary | 75 |
| 7 Conclusions and Future Directions | 77 |
| 7.1 Summary of contributions | 77 |
| 7.2 Challenges and Future Directions | 79 |
| References | 81 |
| List of publications | 94 |

LIST OF TABLES

| Table No. | Table Name | Page No. |
|------------------|---|-----------------|
| Table 2.1 | Overview of HAR methods | 17 |
| Table 2.2 | Comparative analysis of datasets | 19 |
| Table 2.3 | Year-wise comparative analysis of techniques | 21 |
| Table 3.1 | Size of video and segmented video of NTURGB+D dataset | 30 |
| Table 3.2 | Recognition rate per classes of testing data | 37 |
| Table 3.3 | Class wise accuracy, IoU and Mean scores values | 39 |
| Table 4.1 | Name of activities considered for analysis with video label and video label in NTURGB+D dataset | 49 |
| Table 4.2 | PSNR of each video using Farneback, Horn Schunck, Lucas Kanade, and Lucas Kanade Derivative of Gaussian | 49 |
| Table 5.1 | Comparison with other state of state methods | 60 |
| Table 6.1 | Comparison with other methods based on accuracies | 71 |
| Table 6.2 | Activity ID with Activity Name | 71 |

LIST OF FIGURES

| Figure No. | Figure name | Page No. |
|-------------------|---|-----------------|
| Figure 1.1 | Application of Activity Recognition System | 3 |
| Figure 1.2 | Learning family | 6 |
| Figure 1.3 | Difference between DL and ML | 6 |
| Figure 2.1 | Application domain of activity recognition | 13 |
| Figure 2.2 | Sample action “Kick” performed by actors | 16 |
| Figure 2.3 | Example views of five camera | 16 |
| Figure 2.4 | Single from the 8 videos | 18 |
| Figure 2.5 | View of 8 cameras showing sample actions and sample actors | 20 |
| Figure 2.6 | Figure 2.6 Three views in NTURGB+D (side view (+45°), front view (0°) and side view (-45°)) | 23 |
| Figure 3.1 | Steps for action segmentation | 26 |
| Figure 3.2 | Data of Joints Color X and Joints Color Y | 27 |
| Figure 3.3 | Steps for action segmentation per frame | 28 |
| Figure 3.4 | Results of the experiment on the "type on a keyboard" class were shown with an input video (on the left) and a video segmented by action (on the right) | 28 |
| Figure 3.5 | The outcome of the experiment on the “salute” class [illustrated by the input video (on the left) and the action-segmented video (on the right)] | 28 |
| Figure 3.6 | The results obtained from the experiment on the "clapping" class [with an input video (on the left) and a segmented action video (on the right)] | 29 |
| Figure 3.7 | The outcome of the experiment on the "pushing" class [with the input video (on the left) and the video of the segmented action (on the right)] | 29 |

| Figure No. | Figure name | Page No. |
|-------------------|---|-----------------|
| Figure 3.8 | The results of the experiment conducted on the class "drinking" were depicted in the form of an input video on the left and an action-segmented video on the right. | 30 |
| Figure 3.9 | Encoder-Decoder Structure | 33 |
| Figure 3.10 | Steps to test network | 33 |
| Figure 3.11 | The following is a list of 32 object class names, along with the color used to label | 34 |
| Figure 3.12 | Figure 3.12 (a) A sample of a frame taken from the CamVid dataset, (b) An example of a frame labeled on a pixel-by-pixel basis from the CamVid dataset. | 34 |
| Figure 3.13 | Videos with a title, frames per second, the quantity of frames, and length of time are labeled | 35 |
| Figure 3.14 | Plot showing the development of accuracy over time as a result of the training process (iterations) | 36 |
| Figure 3.15 | Plot of loss over the number of iterations during training | 37 |
| Figure 3.16 | The evaluation of trial data, where (a) and (c) are RGB frames from the CamVid dataset, (b) and (d) are the corresponding segmented images, is presented | 38 |
| Figure 4.1 | The activity labels under three categories are medical conditions, Daily actions, and Mutual actions of NTURGB+D | 45 |
| Figure 4.2 | (a) Video-6 (First frame), (b) Video-6 (Third frame), (c) Difference between two frames | 46 |
| Figure 4.3 | Video-6 (Frame-1) magnitude of optical flow a) Farneback, b) Horn Schunck, c) Lucas Kanade, and d) Lucas Kanade Derivative of Gaussian | 46 |
| Figure 4.4 | Video-6 (Frame-1) orientation of optical flow a) Farneback, b) Horn Schunck, c) Lucas Kanade, and d) Lucas Kanade Derivative of Gaussian | 47 |
| Figure 4.5 | Video-6 (Frame-3) magnitude of optical flow a) Farneback, b) Horn Schunck, c) Lucas Kanade, and d) Lucas Kanade Derivative of Gaussian | 48 |

| Figure No. | Figure name | Page No. |
|-------------------|---|-----------------|
| Figure 4.6 | Video-6 (Frame-3) orientation of optical flow a) Farneback, b) Horn Schunck, c) Lucas Kanade, and d) Lucas Kanade Derivative of Gaussian | 48 |
| Figure 5.1 | Process of Activity Recognition System | 52 |
| Figure 5.2 | The LSTM block architecture | 55 |
| Figure 5.3 | Network model for HAR | 57 |
| Figure 5.4 | The HMDB51 contains different categories based on a) body parts that are visible, b) the motion of the camera, c) the point of view of the camera and d) the quality of the video clip. | 58 |
| Figure 5.5 | a) The progress of training using HMDB (10 Classes) is represented by a blue-colored continuous line, b) The changes in loss are displayed by a red-colored continuous line. | 59 |
| Figure 5.6 | a) Utilizing HMDB (51 Classes) for tracking the progression [Blue colored solid line], b) Alterations in the loss rate [Red colored solid line] | 59 |
| Figure 6.1 | 3D Skeleton features using the transversal tree | 64 |
| Figure 6.2 | Step for 3D coordinates of Skeleton 3D NTURGB+D dataset. | 65 |
| Figure 6.3 | Joint location of the human body of Skeleton 3D data [25 Joints] | 65 |
| Figure 6.4 | Architecture of CNN_RNN_1 for feature reduction | 67 |
| Figure 6.5 | Architecture of CNN_RNN_2 for classification | 68 |
| Figure 6.6 | Three views in NTURGB+D (side view (+45°), front view (0°) and side view (-45°)) | 69 |
| Figure 6.7 | Training progress plot (Accuracy versus number of iterations) [Blue line represents training progress and the Black line represents validation progress] | 70 |
| Figure 6.8 | Validation loss plot (Loss versus number of iterations) [Red line represents training loss progress and the Black line represents validation loss progress] | 71 |
| Figure 6.9 | Confusion matrix (Actual Class versus Predicted Class) | 73 |
| Figure 6.10 | Two stream networks of polydomain learning of action recognition | 74 |
| Figure 6.11 | Detailed two stream networks of polydomain learning of action recognition | 74 |

| Figure No. | Figure name | Page No. |
|-------------------|---|-----------------|
| Figure 6.12 | Variation of Accuracy vs number of iterations | 75 |
| Figure 6.13 | Variation of Loss versus number of iterations | 75 |

CHAPTER-1

INTRODUCTION

1.1 Problem Statement

Action is when we are doing something, especially when dealing with anything like an object or human. And activities related to humans are playing significant research areas nowadays. Any human activity recognition system aims to identify and track movements and actions occurring in videos in an automated manner. The ability to identify complex human actions from videos utilize in diverse applications. Identifying human activities allows for real-time surveillance of areas such as airports, train stations, patients, kids and older people.

1.2 Notion of Human Actions and Activities

People take part in a variety of activities. Depending on their complexity, these activities can be categorized into four levels: gestures, actions, communication, and collective activities. Gestures involve body movement, such as kicking or stretching arms. Actions are more complex, like running or driving. Interaction is when two or more people interact with one another, such as by talking or playing games. Group activities involve multiple people interacting with each other for a common goal, like playing soccer or going on a hike. Actions involving signal person activities like “walking” or “waving” may comprise multiple gestures. The interaction involved activities performed by two humans and an object. And it is also called human-object interaction. For example, “two-person fighting”, “stealing a wallet from other people”, “handshaking each other,” etc. And last but not least, Group activities perform by the conceptual group of persons or multiple persons or objects like “A group of people marching”, “a group of having a meeting”, or “two group fighting”[1].

Computer vision is an interdisciplinary field. Computer vision motivates us to automatically extract, analyze and understand single or series of images. Computer vision is a technology increasingly use in industry, science, law enforcement, consumer protection, and general purpose to detect human activity from video. This technology has opened up a new realm of possibilities for its application in these various fields. In the mid-19th century, the pioneering photography of E.J Marey and E. Muybridge revealed remarkable insights into the dynamics of human and animal movement. Subsequently, Johansson explored this further with his moving light display experiment [2]. At the onset of visual vector analysis, the mechanics of motion perception were examined, with the findings applied to biological motion patterns. The experiment conducted

was used to assess the accuracy of this context. This paved the way for developing human action and automatic recognition models, which have begun to be useful in computer vision.

When it comes to the area of recognition or tracking, surveys related to vision often use words such as "actions" and "activity". An action is a simple movement performed by an individual for a brief period of time, such as "raising a hand," "bending," or "swimming," etc. On the other hand, activity is a longer and more complex movement, usually involving multiple people or objects. But, the activities concern a complex sequence of actions accomplished by many humans involved each other in them in some manner. These actions are classify as longer-term activities, such as two people shaking hands, a soccer team scoring a goal, or a group of individuals commandeering an airplane, stealing, etc. [3].

1.3 Applications of activity recognition

The utilization of vision-based activity recognition systems has had a significant influence in many inspiring application areas, including biometrics based on behaviour, video analysis based on content, security and surveillance, interactive applications and environments, animations, and syntheses [3]. Behavioral biometrics involves approaches based on Fingerprint, Face or Iris and use to recognize human-based physical or behavioral cues. In this approach, the subject corporation requires and only gets to know the subject activity. Gait recognition [4] could be the most challenging application area because a person walking characteristics can identify the person through CCTV (closed-circuit television) footage. However, everyone has a distinct walking style like other biometrics. With the speedily advancing technology, it is becoming easier for people to share and look up multimedia content, such as images, music, and videos. However, searching for desired content remains a daunting task. This is why a retrieval system has been developed to help locate a selection of objects with similar content [5]. Summarizing and retrieving consumer content, such as general activities like sports videos or cooking videos, are the most commercially applied under content-based video analysis. A new visual monitoring system can monitor the movement of objects in an area and acquire knowledge of the activity patterns from the actions. This system consists of motion tracking, activity categorization, and event detection.

A site can be large to observe from a single camera so many such sensor units distribute around the area. Cameras attach to poles, trees, and buildings for an outdoor setting. The indoor environment involves attaching to walls and furniture [6]. Research into intelligent surveillance has become more prominent due to its successful implementation in public areas such as airports, railway stations, shopping malls, crowded places, military installations, and smart healthcare facilities. Its purpose is to identify, recognize, or learn about activities that could be classified as

“suspicious”, “irregular”, “uncommon”, “unusual”, “abnormal”, or “anomalous” [7].



Figure 1.1 Application of Activity Recognition System

For such activity, using CCTV cameras to record or observe scenes the user has become ubiquitous. Although recording videos through cameras are cheap, affordable, and popular today. However, the agents for observing outliers and analyzing the footage are limited and not reasonable also. Wherever video cameras use for observation, a person's monitoring of activities is not genuinely done for reasons like fatigue. The operator feels bored between the scenes because the time duration to happen any suspicious activity is short, just a few seconds or nothing that catches the alarm in the scenes. This application comes under security and surveillance because detecting unusual activity at the right time is an essential task. In interactive applications and environments, the interaction between human and computer is one of the challenges of designing a human-computer interface. An interactive environment such as smart rooms that can show a response to person gesture can benefit directly or indirectly to the user. Such that, on music according to the user's mood when entering the room. Animation and synthesis, where requires a large collection of motions used by the animator to make high-quality animation or movies shows figure 1.1. Any application can relate human motion to any environment, including training military soldier, firefighter and other personnel.

Falls are a major health concern, due to their significant impact on survival rates. The World Health Organization reports that every year, between 28-35% of people aged 65 or older and 32-42% of those aged 70 or older experience a fall. As such, fall detection is a critical issue that needs to be addressed [8]. As one gets older, this statistic rises, and for those aged 79 or more, falls are the primary reason behind death caused by injuries. The 0.6 million older people in the

United States suffer from fall-related injuries annually. As the elderly population grows, more are living alone in private dwellings. For those who are not found for more than an hour after a fall, death can occur within six months, even without direct injury. This calls for developing an intelligent system for seniors that can automatically detect falls and alert family members or caretakers in a timely manner [9].

1.4 Notion of Activity Recognition System

A behavior or activity recognition system typically involves a series of steps. According to [3], this includes introducing a video or sequence of images, extracting descriptions of functions and actions, and finalizing by interpreting the action. Despite advancements in the area, human activity recognition is still a difficult research problem due to the existence of large intra-class variations due to the speed and motion pattern, viewpoint, and background clutter; the correlation between action recognition and complex circumstantial information such as scene and object characteristics; and the variety and changing nature of an action category, which can make it difficult to model primitive and short-lived actions between consecutive actions [10].

In single-view learning, a model is learned for each action from a single static camera view, and then data that has not been seen is classified as one of the classes. An alternate to single-view learning is cross-view learning, which is a way of mapping the features from various perspectives into one unified feature space. This is done to address the disparities in visual appearance that this view does not account for, such as differences in appearance and motion attributes from other camera views. This type of learning is part of the four machine learning problems that fall under action recognition, including cross-domain and multi-task learning.

Multiview and multimodal description contains a definite gap between object representation [10]. The same target may have a different description from the multi-view observation space, which is majorly correlated but sometimes looks different from each other. Observing that each outlook of the data has a certain understanding that other perspectives lack, indicates that multi-view learning can be more comprehensive and articulate than single-view learning. Although, there are so many algorithms have been proposed, but still, some problems are in that algorithm. Current methods focus on single-view learning, which lacks relationships among multiple views. Then, the current approach ignores consistency between different views [11]. This concept illustrates that when different individuals enact the same scenes or events, there will be differences in posture and rate of movement, as well as differences in the environment in which the action occurs, leading to different visuals, backgrounds, camera movements, lighting, and occlusions. Creating systems for action recognition that are able to recognize all variations within

one class and distinguish between activities of various classes is a major task. A major emphasis must be placed on creating discriminative features from data to recognize actions effectively. The representation of the feature can be broken down into two groups: global and local representation. Global representation considers the entire observation and can be derived from silhouettes, trajectories, optical flow and edges. It is possible to view the observation as a gathering of descriptors obtained from temporal and spatial focal points [12].

The researcher needs datasets to create a classification system for actions. Data is collected in a dataset that can be 2D or 3D, and each dataset has its own issues such as resolution, frame rate, actions/actors, background, and application domain. Depending on the dataset, it can contain different characteristics like ground truth, number of actions/actors, views, and application areas. Developing an effective algorithm is very important rather than collecting data about the dataset. After the emergence of low-cost depth sensors such as Microsoft Kinect, Asus Xtion, and Prime Sense, researchers prefer 3D and multimodal video datasets more. Low-price sensors can be employed to acquire video footage which includes multiple components such as depth frames, accelerometers, IR sensor frames, acoustical data, and skeleton data. RGB-D images or videos are more advantageous than RGB images or videos, as they are less vulnerable to alterations in light intensity, blockages, and background messiness. In this context, RGB and RGB-D information are mutually beneficial [13].

1.5 Deep learning

DL is an algorithm used in machine learning that allows a computer system to interpret and analyze complex data sets, as well as make prognoses about what could happen in the future. This technology uses to design a computer model that can act, learn and interpret the same as human intellectual skills. Nowadays, this is the key technology behind autonomous cars, which recognize various signs on the road and help the driver to safe driving. Deep learning is a branch of machine learning (ML), and ML is a subfield of artificial intelligence (AI), as illustrated in Figure 1.2. AI is a program that senses, reasons, acts, and adapts like human intellectual sense. Under that, ML consists of algorithms that perform various tasks after data exposure over time [14]. In the last DL, many-layered networks learn and do classification or regression. This thesis is based on classifying various activities using deep neural networks (DNN).

If comparing the ML and DL, the ML is a conventional technique that requires various steps, as given in figure 1.3. The steps are preprocessing features extraction and feature selection in the last classification. In comparison, DL can provide output in one shot. Its self-DLL consists many numbers of the layer, which help to learn appropriately as per the input and classify image

or video as per labeled data. To get more recognition rate, it's always necessary to know the system in some pattern of input features. That feature calculation can use if the data is enormous. So that system can analyze and classify more rates.

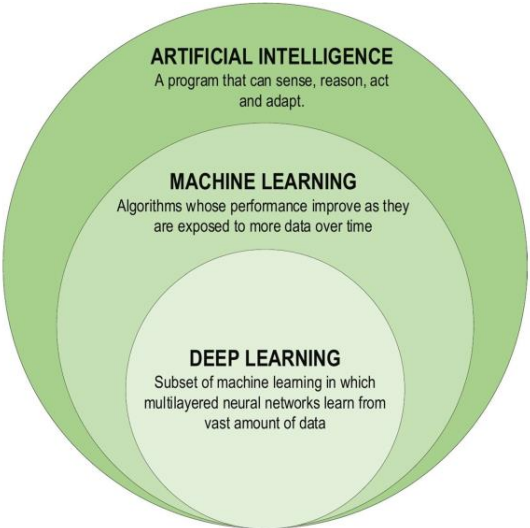


Figure 1.2 Learning family [14]

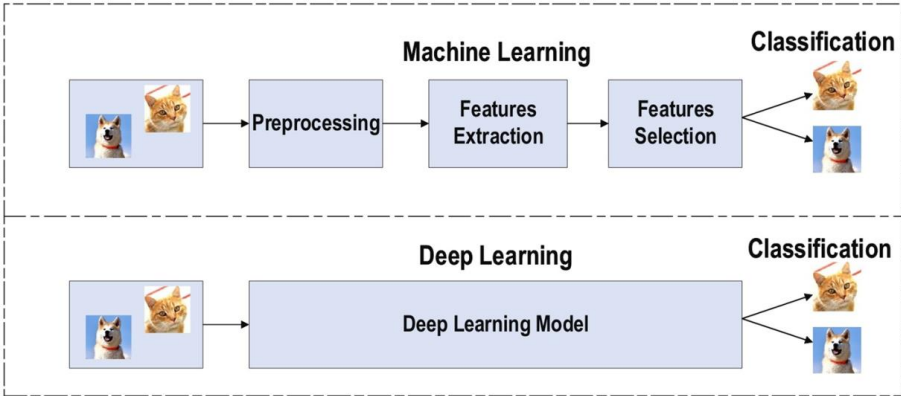


Figure 1.3 Difference between DL and ML [14]

1.5.1 Polydomain learning (PDL)

Identifying human activities is a critical element of computer vision research and applications, especially when bringing together different viewpoints and forms of information for use in a variety of practical applications, from human-computer interaction to advanced video surveillance and understanding and managing of multimedia content. Research into detecting actions from varying perspectives and forms is still ongoing. There is a requirement of the system which recognizes activities after PDL. And the availability of the video sequence database is huge, attracting researchers to do research in this field. This work collectively defines many views and many modals as PDL.

1.6 Data modalities

There is a different modality in which research is being gone to identify human-related activities in the videos. There are four modalities which are as follows:

- a) *Depth Map*: The depth maps are sequences of two-dimensional depth values in millimeters or depending on the dataset. Compared to the RGB, the impacts of changing lighting, intricate backdrops, and camera movement can be detrimental to these elements. Compared to RGB images, the depth is not as susceptible to fluctuations in lighting, obstruction, and interference from the backgrounds. The depth map image sequences are also called the RGB+D database.
- b) *3-dimensional*: Human pose can represent in an articulated arrangement of joints and limbs, known as skeleton data. This type of data is widely utilized for action recognition. In contrast to RGB videos, skeleton data furnishes an exact account of actions. So, it is more suitable for human activity recognition.
- c) *Infrared sequences*: Infrared action detection has received more and more attention recently due to the robustness of infrared images against color and lighting changes and cluttered backgrounds, and it works well even in low light. Neural networks have had remarkable success in visible-light-based human behavior detection but cannot be directly applied to infrared behavior detection because infrared images lack color and detailed appearance information [15].
- d) *RGB*: Most common modality is RGB. RGB videos contain 2D data only, which has information for the height and width of the object.

In this research, only two modalities are used to design a system that recognizes activity.

1.7 Notion of research gap

As per the literature survey mentioned in Chapter 2, the following points is required to investigate:

1. In the experimental work mentioned in Chapter 2, the model requires deep learning, which can handle more classes. In this work, the final model for activity recognition can take 60 classes of dataset NTURGB+D. Further details are presented in chapters 5 and 6.
2. Designing a model that can take input in different modalities like RGB and skeleton 3D is required. In the same way, information can have many three views in this experimental work. Further details are presented in chapters 5 and 6.

1.8 Thesis contribution

The thesis has contributed to the field of AI-based human action recognition systems in the following ways:

- *Segmentation methods for RGB video frames*

The method has been given, which can segment subjects from RGB frames using Skeleton 3D information using the NTURGB+D dataset. The mentioned contribution is presented in the chapter in chapter 3. Object recognition using semantic segmentation is also presented in Chapter 3. And this part of the work, two papers were published.

- *Feature extraction for object detection*

The feature extraction for object detection is elaborated in Chapter 4. This part of the work was published based on an analysis of four methods for feature extraction of object detection.

- *Activity recognition framework, its evaluation, and analysis*

The BiLSTM network for action recognition using transfer learning is presented in Chapter 5, and this part of the work was published.

The action recognition framework for Multiview is presented in Chapter 6. The contribution of chapter 6 is as follows:

1. Transversal tree-based representation of skeleton 3D information
2. Deep neural network that can handle Multiview data
3. Network which can act as a feature reduction method
4. Deep neural network for classification of action

1.9 Objectives of the study

The following main objective is defined to fill that research gap as given below:

Objective 1: To extract the Poly domain features from video frames.

Objective 2: Reduction of extracted features using feature reduction method.

Objective 3: Development of an algorithm for human activity recognition with validation using benchmark data set.

As per objective 1, the various features analyzed, like HOGs and HOF, are mentioned in Chapter 4. And, also worked on preprocessing techniques like segmentation refer to Chapter 3. Finally, the representation of input, i.e., videos having more than one view, is presented using the transversal tree.

As per objective 2, various ways have been given to reduce features. Chapter 3 introduces subject segmentation of actions of RGB frames. Chapter 5 presents the representation of videos in sequences using transfer learning. In Chapter 6, a neural network to find optimal features is

presented. These ways can be used to represent input data in various forms.

As per objective 3, The frameworks that can classify actions from RGB videos using machine learning is presented in Chapter 5 with the concept of transfer learning. In Chapter 6, the framework that can classify activities from input (Skeleon 3D) using deep neural networks. The polydomain learning model is presented, which can take input as skeleton 3D and RGB videos of 3 views.

Activity detection aims to discover one or more agent behaviors and goals from observations of agent behavior and environmental conditions. In the past four decades, there has been considerable progress in this field of study, which has been a significant source of comprehensive, personalized assistance for various applications. This field of research has connections to multiple subject areas such as sociology, medicine, and human-computer interaction. It has become a topic of focus for several computer science circles. As per chapter 2, The important points to fill the research gap in the human activity recognition field have been covered.

1.10 Thesis structure

This document is divided into chapters to organize this research work into coherent parts. The detail of the chapters as follows:

- *Chapter 2-* Literature Survey

This chapter provides a detailed survey related to human action recognition methodologies. The year-wise analysis of methods, noteworthy key points, various datasets, and evaluation parameters considered by researchers.

- *Chapter 3-* Subject segmentation of actions from RGB frames with skeleton data

In this chapter, we proposed a related technique for action segmentation of RGB video frames. The data set for approach evaluation is the NTURGB+D data set. This approach may require the segmentation of interest from each video frame.

- *Chapter 4-* Feature extraction for object detection

In this chapter, the various features evaluate, i.e., HOF and HOG. The HOF (Histogram of optical flow) is one of the preferred features for action recognition as per the literature survey. It is essential to understand the motion of the subject. And HOG (Histogram of oriented gradient) also explores object detection in the video frame.

- *Chapter 5-* Human Activity Recognition using BiLSTM Network

We propose the human activity recognition model using BiLSTM Network in this chapter. The network design with the concept of transfer learning. Each video represents the

sequence vector using CNN (Googlenet). The sequences input gives to the BiLSTM network for action recognition. The HMDB51 dataset uses for approach evaluation.

- *Chapter 6- Action Detection in Multiview Skeletal 3D Data Using NTURGB+D Dataset*
We propose an action recognition model for Multiview using skeleton data. This work presents the technique to calculate the optimal feature calculation. And it also presents a model which recognizes the actions.
- *Chapter 7- Conclusions and Future Direction*
In this chapter, we present an overview of the key advantages and disadvantages of the suggested approaches, along with possibilities for future development, extensions and outlooks of the proposed approach.

This chapter includes the basic concept of human action recognition. Also, elaborate on how action recognition is helpful with various applications. Action recognition is an essential aspect of dealing with multi-domain learning. The next chapter will detail a literature survey based on human action recognition with their method and performance.

CHAPTER-2

LITERATURE SURVEY

Studying the most recent advancements in activity detection can be quite captivating. There is the various application of activity recognition. The researcher has more inclination toward intelligent systems for better human and computer interaction, such as controlling the presentation slides through hands or cognitive steps that help workers learn and enhance their capabilities. As an illustration, systems with a memory-based attention module can be employed to create designs for large public areas [16]. In today's scenario, activity recognition uses for assisting disabled people, older adults, and regular people. Monitor multiple individuals and personalities in a specific nursing home using orientation-based algorithms [17]. It also has the advantage of drivers for assisting as an intelligent driver assistance system like modeling for driver behavior, awareness or predicting driver turn and in animation also. The development of autonomous mental abilities through recognition of real-time interactions with the environment using sensors and effectors is also part of its application in human activity detection.

Now a day, everybody is more toward smart things to make their own life comfortable, so the smart environment plays an important role like Observing the concentration and involvement of participants in the discussion room. In sports, activity recognition is applied, e.g., by analyzing performance and training. One of the most important aspects of society is surveillance means raising the alarm for anomaly detection in many places such as buildings, critical infrastructure, public transportation, parking a lot, homes and airports etc., or at a location open to the public will be required. Video annotation is also part of motion-based activity recognition for outdoor sports broadcasts [18]. All the application domains broadly have an application under activity recognition, as shown in figure 2.1

2.1 Survey of HAR System

This chapter includes the survey of the current knowledge, essential findings, and theoretical and methodological contributions of the topic. The related work can be divided into multi-view representation and multimodal representation. The former knowledge is given to multi-view and multi-modal sources. There are various methods have been developed for view variant action recognition. It is possible to group these techniques into three distinct categories: the first being view-independent techniques used to develop the necessary classifying systems. Classifying can be accomplished through two different ways: using multiple classifiers for various categories or using one singular classifier with training data from multiple perspectives. The second type of methods are cross-view actions like training using single view and classifying or recognizing

action from all the views [19]. Different strategies have been utilized to handle the second type of issue, for example, transferable dictionary learning, specificity and latent correlation learning. Two different dictionary learning approaches for the transferable learning method can be applied to obtain the sparse representation of videos. In the first approach, the corresponding videos in a set are given the same sparse representation by establishing view-specific dictionaries. The other approach creates a common dictionary and view-specific dictionaries for reconstruction. The author has done analysis using these two approaches with three datasets IXMAS, WVU and MuHAVI and accuracy show 97.8%, 98.9% and 98.5% respectively [12]. In the field of cross-view recognition, a limitation manifests in the form of testing video views not being fused with knowledge from other views. As an alternative, specificity and latent correlation learning can be used for cross-view action recognition. In [20], proposed method using synthetic data from depth maps, uses dictionary view specific and considers correlation between different views. The researchers utilized Gesture3D, MSR action3D, ChaLearn, and action pairs multi-modal dataset in their experiment and recognition rate was 95.77%, 98.42%, 95.43% and 91.3% respectively. In the future work, they will use deep learning for multiview action recognition.

Third type methods are view invariant action representation for action recognition. In [21], present view invariant method and recognize action by considering motion and shape based information through HMM. This algorithm is robust to variation in view and duration but not dealing with complex activities. To prove robustness, author used own recorded video clips and correct recognition rate is 88.3%. They have not proven the things on any standard datasets. For human activity recognition from image sequence using independent component analysis for feature extraction, code book generation and recognition using HMM shows accuracy of 97.50% because only 10 classes has been taken [22]. A method of view-invariant action recognition which uses an artificial neural network has been suggested as a possible resolution to the problem of action recognition. In [23], the main contribution is using SOM to identify the basic posture of actions and use fuzzy distance to attain invariant action representation. The Bayesian framework used to create for recognition results of multiple views. In experimental setup, the author used two standard datasets i3DPost and IXMAS. They got accuracy 97.8% using i3DPost and 89.8% using IXMAS. Another approach using SOM given by Maddalena et. al [24] can handle videos with moving background, illumination etc. Some view invariant method is obtained by circular invariance property of DFT also. And, used discriminant analysis for dimension reduction for calculated features. And, according to author the recognition rate will be more and approach will performed fast action recognition [25]. The recognition rate for action is 96.34% using dataset of i3DPost. In [26], the application of the Discrete Fourier Transform (DFT) in combination with

Fuzzy Vector Quantization (FVQ) and Linear Discriminant Analysis (LDA) for the classification of posture images from multiple views has been demonstrated to yield an accuracy rate of 90% using an in-house database.



Figure 2.1 Application domain of activity recognition [18]

To find correct features, some researcher has been used self-similarity matrices for creating more flexible dictionary with collaborative sparse coding and action label prediction. In this way, accuracy recognition rate was not less 79.18% using IXMAS dataset [27]. However, those are not towards optimal feature calculating using previous ways. They can prefer deep learning approaches. The method using CNN framework is given by [15], this work efficiency proved by considering two datasets InfAR and NTU RGB+D and recognition rate is 79.25% for first data and second is 66.29%. NTU RGB+D is having 60 classes and having different modality also. The accuracy rate is less due to this work needing enhancement in features calculation and only did work for infrared human activity recognition.

Human activity recognition has different approaches, which also apply 2D multiview images and 3D based techniques. In 2D approaches, action recognition included movement of body and level of gestures. In motion features of 2D approaches, some authors perform action recognition using motion-based feature from image sequences in different view angle. Shape and

silhouette features-based recognition methods for classification with efficient sequence matching algorithm [28] shows 91.2% using Weizmann dataset. In this methodology, no account is taken of viewpoint invariance and orientations.

Ahmad *et.al* [29] use the KU dataset and apply PCA to the optical flow velocity and human shape information to create a set of HMM for each action and viewpoint to represent the actions, it shows accuracy of 87.5%. In [30], author combine local and global features optical flow to extract features and represent each action with view point using multidimensional HMM then accuracy shows 88.33%. Features have as a prominent attribute for classification process. There is various method used for calculating to improve classification correction rate. In [31], used low dimensional feature vector based action and includes dynamic warping for recognition shows 80.05% accuracy with IXMAS dataset. Similarly, to improving accuracy work has been going on multiple features to capture view variation, illumination and attributes. Some work has been going on system designing that system can use with any type of features [32]. In [33], involves motion features and two steps for recognition: using local features by nearest neighbor and using simple strategy to label actions. This strategy has shown accuracy as 90.3% using KTH which has 6 classes only.

The utilization of an index into a multidimensional hash table to recognize activities based on views is presented in [34]. It is possible to recognize the activity in a video sequence by examining only a few frames. An approach considers set of pose and velocity vector for major body representation in hash table. This approach shows accuracy of 84.6% but only consider three view. Author has proposed various methods for multiview action recognition. In [35] has given a view specific model called latent variable discriminative model which has advantage over single view approach. This model can extend to work with data where views are not defined. An approach based of Bag of word and naïve Bayes nearest (NBNN) used for action recognition has accuracy of 88% using IXMAS dataset[36]. In [37], developing a hierarchical partwise bag of word representation for single and multiple view human action recognition, and using three databases for analysis. From which maximum accuracy is 96.7% using KTH dataset. For multiview recognition, graph model can also use after extracted space time interest point of each view [11] gives accuracy rate of 97.94% using IXMAS dataset. In [38], used Cauchy estimator feature embedding presented of depth images and compared classical method with various method PCA, LPP , LDA and DLA shows 97.4% accurate. In [39], 2DPCA has been proposed for both spatial and transform domain and in future, will apply using multi-transform shows accuracy of 98.99% with Weizmann dataset. If action recognition using Latent Kernelized SVM then accuracy using IXMAS is 97.09%. Not everyone deals with unlabeled data, method convex multiview semi

supervised classification has given accuracy of 59.08%. Proposed method based on scene flow estimate using RGB and depth data for 3D action recognition has been given accuracy of 87.6% using M2I [40]. A recognition algorithm was developed that utilized ground subtraction from frames and motion/texture information of each pixel, which was recorded in binary. This yielded an accuracy of 95.5% when tested with the i3DPost dataset. In [41], attention based multiview fusion model has been proposed for skeleton action recognition having accuracy of 95% maximum using SBU Kinect interaction. To reduce dimension of matrices is important aspect in appropriate feature selection. Author works using sparse low rank representation and handles missing values also shows accuracy of 84.62% [42]. Sparse based regression model of multimodal features of depth and skeleton based features used for action recognition with weight regularization having accuracy of 100% using 3D action pairs (having only 6 classes) [43]. Some work has been going on bag of features and SVM by creating flexible dictionary and robust classifier having accuracy of 93.9% using UTL Action3D.

The various deep learning frameworks explained by author Hasan et. al [44] has been given hierarchical feature model in which best feature separately calculated. After doing this, the best accuracy shows 98% using KTH dataset. In KTH dataset, the action classes are only six. Author has shown experiment on other dataset also in which classes are more as compared with KTH then minimum accuracy is 53.8%. In [45], proposed approach based on discriminant feature fusion framework for RGBD and also consider the inter and intra modalities correlation with accuracy shows 41.5% only. In [46], proposed weighted hierarchical depth motion maps and three channel network using action recognition system and this method can be benefit when have cluttered background. Authors also commented that if action classes increase, then performance will decrease. The experiment setup has shown by using four standard datasets and a maximum accuracy of 94.95% for the MSRAction3D dataset. In [47], present deep learning based framework for action recognition using RGBD with structured based classifier. Author used different dataset to show efficiency of own model from which maximum accuracy is 100% on 3D action pairs. There are various modalities as RGB, RGBD, skeleton and Infrared. In [48], authors have presented model for 3D skeleton, body parts images and motion history images based hybrid model using a convolutional neural network. Author will extend the work by including other modalities in work. The model analysis has been done using various datasets from which maximum correction rate is UT-Kinect i.e. 99.2%. In this dataset the number of classes are only 10, however author did analysis on dataset where classes are 60. Using NTU RGB+D dataset, accuracy shows 90.4%, in this model also has limitation that is if number of classes increased the performance would have decreased [9]. Human activity recognition from red green and blue (RGB), depth, skeleton sensor

data has drawn increasing attention. Multiview learning is model learning from multiview data of different views has been important vision-based task. In [49], explores the use of deep learning for action recognition based on skeleton data and accuracy shows of maximum 97.62% using MSRC-12 in which classes are 12 only, however if consider Recognition using database NTU RGB+D shows 90.12%. The comparison of these method is given in table 2.1.

In Table 2.2, a comparison of various reviewed approaches to human action recognition, both qualitatively and quantitatively, has been demonstrated using a variety of datasets. An examination of human activity recognition was conducted using publicly available datasets such as the IXMAS Multiview Human Action Dataset [50], i3DPost multiview human action and interaction dataset [51], MuHAVi (Multi-camera human action video dataset) [52] and NTU RGB +D [53]. Using the IXMAS dataset, the maximum accuracy evaluated is 97.94% [11] . The author has conducted single view and multiview learning after obtaining space time interest points for each view, then employed the multiview bag of words representation and utilized a graph model. The IXMAS (Inrai XMAS motion acquisition sequence) aims to form state-of-the-art action recognition. This dataset contains 11 actions (*pick-up, walk, turn-around, get-up, sit-down, wave, punch, kick, Check-watch, Cross-arms, scratch-head*), total 10 performers involved in the activity, with five male and five female actors. shows in figure 2.2. the actors altered their orientation, to record the multiview and view invariant data and labels are given. And, the acquisition was done by using five cameras setup [50] shows in figure 2.3.

In i3DPost multiview human action and interaction dataset, eight camera setups were employed to create high definition multiview videos.

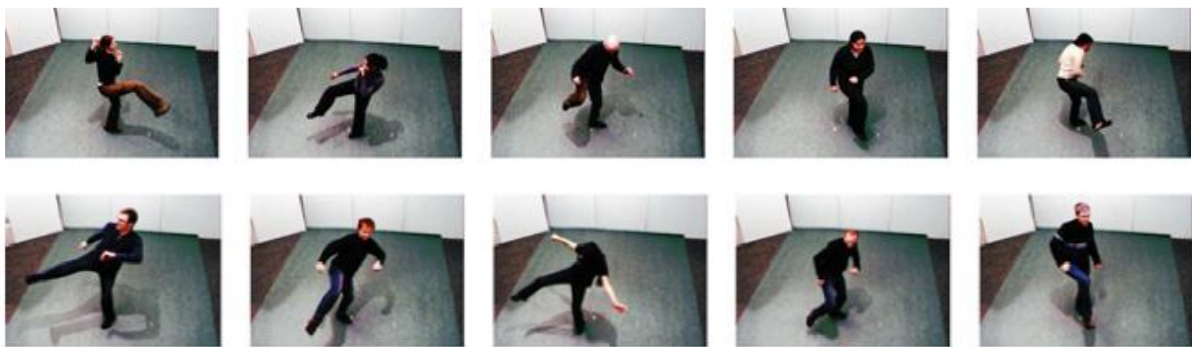


Figure 2.2 Sample action “Kick” performed by actors



Figure 2.3 Example views of five camera

Table 2.1: Overview of HAR methods

| Author, Year | Methodology | Dataset (#videos/sample/sequence, #actions, #subjects, #views, #modality) | Parameters (Average Accuracy (Acc. %)) |
|---|--|---|--|
| Yale Song, Morency, and Davis 2012 [35] | Multi-view latent variable discriminative | ArmGesture(120 samples,6,13) | ArmGesture (97.65 %), |
| Gao et al., 2015 [11] | Multi-view discriminative and structured dictionary learning with group sparsity and graph model | CVS-MV-RGBD single (4320 sequence, 10, 18, 3, RGB/Depth/Skeleton), IXMAS (1815, 11, 11, 5, RGB) | CVS-MV-RGBD single (92.19%), IXMAS (97.94%) |
| Hasan and Roy-Chowdhury 2015 [44] | Deep Learning based hierarchical feature model | KTH (599, 6, 25), UCF50 (6676, 50, 25), VIRAT (NA, 11), TRECVID (NA, 7) | KTH (98%), UCF50 (53.8%), VIRAT (62.8%), TRECVID (66.65%) |
| H. Zhu, Weibel, and Lu 2016 [45] | Deep convolutional neural network | SUN RGB-D (47 scenes, 19) | 41.50% |
| P. Wang et al. 2016 [46] | Deep convolutional neural network | MSRAction3D (NA, 20, 10), MSRAction3DExt (1379, 20, 13),UTKinect-Action (NA, 10, 10), MSRDailyActivity3D (NA, 16, 10) | MSRAction3D (94.95%), MSRAction3DExt (94.35%), UTKinect-Action (92.93%), MSRDailyActivity3D (80.83%) |
| Guo et al. 2017 [38] | Multiview Cauchy estimator feature embedding | CAS-YNU-MHAD (NA, 10, 10, NA, 3-D-acceleration/depth, and RGB) | 97.40% |
| P. Wang et al. 2017 [40] | Scene flow to action map and ConvNets | ChaLearn LAP IsoGD (47933, 249, 21, NA, RGB-D), M2I (1760, 22, 22, 2, RGB) | ChaLearn LAP IsoGD (36.27 %), M2I (87.6%) |
| Shahroudy et al. 2018 [47] | Deep learning | Online RGB dataset (336, 7, 24), MSR DailyActicity3D (320, 6, 10), 3D action pairs (360, 6, 10),NTU RGB+D (56880, 60), RGBD-HuDaAct (1189, 13), | Online RGB dataset (94.6%), MSRDailyActicity3D (97.5%), 3D action pairs (100%), NTU RGB+D (74.9%), RGBD-HuDaAct (99%), |
| El-Ghaish et al. 2018 [48] | Convolutional neural network | UT-Kinect (199, 10, 10, Skeleton/RGB), SBU Interaction (282, 8, 7, skeleton/RGB), Folerance3D Action (215, 8, 10, skeleton/RGB), NTU RGB+D (56880, 60, 40 skeleton/RGB) | UT-Kinect (99.2%), SBU Interaction (99.3%), Folerance3D Action (96.5%), NTU RGB+D (90.4%) |
| Y. Liu et al. 2018 [15] | Convolutional neural network | InfAR (600, 30), NTU RGB+D (420, 10) | InfAR (79.25%), NTU RGB+D (66.29%) |
| Li et al. 2019 [49] | Deep network | NTU RGB+D (56578, 60, 40, 5), UTD-MHAD (NA, 27, 8), MSRC-12 (594, 12, 30) | NTU RGB+D (90.12%), UTD-MHAD (95.58%), MSRC-12 (97.62%) |
| Fan et al. 2019 [41] | Multiview fusion process | NTU RGB+D (56880, 60, 40, 3, Skeleton), SBU Kinect Interaction (230, 8, 7, 1, Skeleton) | NTU RGB+D (85.9%), SBU Kinect Interaction (95%) |
| Xie et al. 2021 [54] | Graph convolutional networks | NTU RGB+D (56880, 60, 40, 3, Skeleton), Kinetics(300000, 400) | NTU RGB+D (Cross subject(88.87%), cross-view(95.33%)), Kinetics(58.9%) |

The video collection in the database includes footage of eight individuals (2 female/6 male) and includes twelve actions (*One person pulls another, two persons handshaking, run-jump-walk,*

run-fall, walk-sit down, sit down-standup, one hand wave, walk, jump in place, jump forward, run.) which are captured from eight cameras as shown in figure 2.4. Using this database, the maximum accuracy under literature review is 97.8%. In this work, the author has given view invariant action recognition using neural network [23].

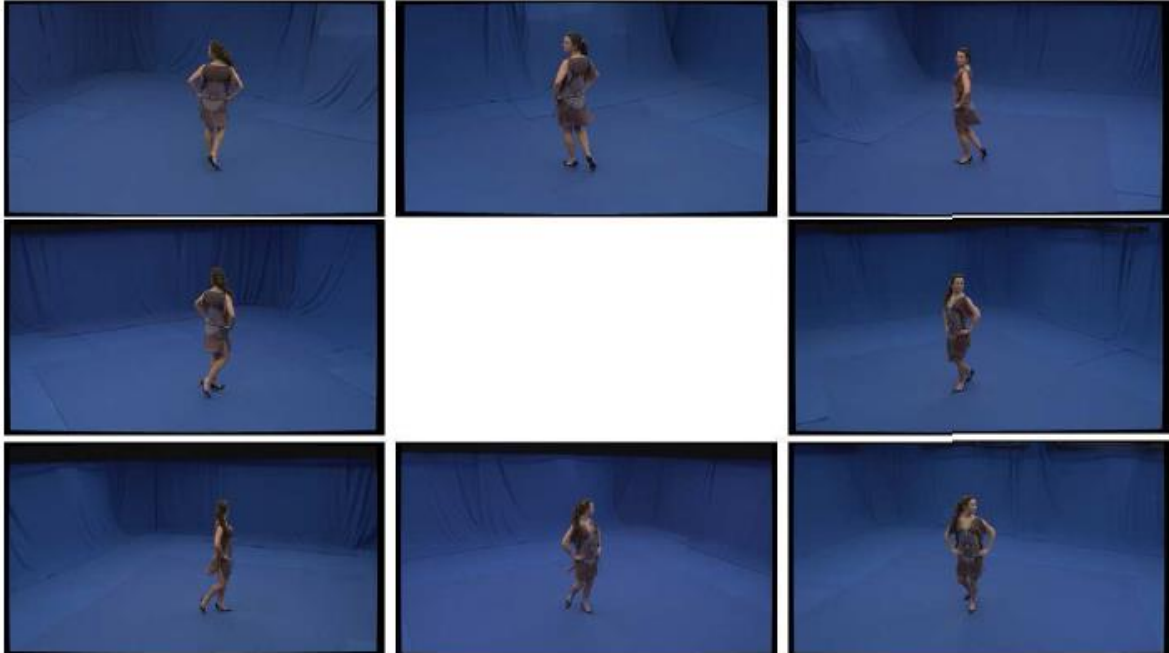


Figure 2.4 Single from the 8 videos

Multi-camera human action video dataset (MuHAVi), another database that researchers prefer to use when multiview action recognition. In this dataset, there are 17 actions (*Walk, Run, Punch, Kick, Shot Gun, Pull Heavy Object, Pickup Throw Object, Walk Fall, Look In Car, Crawl On Knees, Wave Arms, Draw Graffiti, Jump Over Fence, Turn Back, Stop, Collapse, Drunk Walk, Climb Ladder, Smash Object, Jump Over Gap*) performed by 14 actors and captured through 8 Cameras as shown in figure 2.5 [52]. And, last but not least NTU RGB+D database having data modalities, action classes, subjects and views. The compilation of the database comprises four data modalities - depth maps, 3D joint information, RGB frames and IR sequences - and 60 action classes delineated into three clusters: 40 daily activities, 9 actions linked to health, and 11 mutual actions carried out by 40 participants ranging in age from 10 to 35. And, this database contains only 3 views [53]. The maximum accuracy under review of literature is 90.4% using a convolutional neural network that uses 3D skeleton, body part images and motion history images [48]. In future, author will include more modalities to extend the work.

Table 2.2 Comparative analysis of datasets

| References | Dataset | #videos/sample/sequence | #actions | #subjects | #views | #modality |
|--|----------------------|-------------------------|----------|-----------|--------|---------------------------------|
| [53] & [47] | 3D Action Pairs | 360 | 6 | 10 | - | RGB/Depth/Skeleton |
| [55] | ACT42 | 6844 | 14 | 24 | - | - |
| [34] | Activity database | 92 | 8 | | 7 | - |
| [35] | ArmGesture | 120 samples | 6 | 13 | - | - |
| [55] | Berkeley MHAD | 550 | 11 | 12 | - | - |
| [56] | BREAKFAST | 1989 | 10 | - | - | - |
| [38] | CAS-YNU-MHAD | - | 10 | 10 | | 3-D-acceleration/depth and RGB) |
| [7] | ChaLearn LAP IsoGD | 47933 | 249 | 21 | | RGB-D |
| [20] | ChaLearn multi-modal | 13858 | - | - | - | - |
| [11] | CVS-MV-RGBD single | 4320 sequence | 10 | 18 | 3 | RGB/Depth/Skeleton |
| [48] | Folerance3D Action | 215 | 8 | 10 | - | skeleton/RGB |
| [57] | HMDB51 | 6766 | 51 | - | - | - |
| [23], [25]& [58] | i3DPost | 80 | 8 | 8 | 8 | - |
| [15] | InfAR | 600 | 30 | - | - | - |
| [31], [32], [36], [33], [19], [39], [59], [23], [28], [11], [12], [27], [58]& [42] | IXMAS | 2520 | 14 | 12 | 5 | RGB |
| [57]& [54] | Kinetics | 300000 | 400 | - | - | - |
| [33], [37] & [44] | KTH | 599 | 6 | 25 | - | RGB |
| [30] | KTHDB | 600 | 6 | 25 | - | - |
| [29] | KU gesture database | - | 5 | | 8 | RGB |
| [30] | KUGDB | - | 14 | 20 | 3 | - |
| [40] | M2I | 1760 | 22 | 22 | 2 | RGB |
| [37] | MSR action 3D | - | 20 | 10 | - | Depth |
| [20] | MSR Action Pairs | 360 | - | - | - | - |

| | | | | | | |
|---------------------------------------|----------------------------|-----------|-----|----|---|--------------------|
| [43]& [20] | MSR Action3D | 567 | 20 | 10 | | RGB/Depth/Skeleton |
| [20] | MSR Gesture3D | 336 | - | - | - | - |
| [49] | MSRC-12 | 594 | 12 | 30 | - | - |
| [28]& [12] | MuHAVi | - | 17 | 7 | 8 | RGB |
| [37] | MV-TJU | 7040 | 22 | 20 | 2 | RGB/Depth/Skeleton |
| [47], [48], [15], [49], [41]& [54] | NTU RGB+D | 56880 | 60 | 40 | 3 | Skeleton |
| [47] | Online RGB dataset | 336 | 7 | 24 | - | - |
| [47] | RGBD- HuDaAct | 1189 | 13 | - | - | Depth |
| [48] | SBU Interaction | 282 | 8 | 7 | 1 | skeleton/RGB |
| [57] | Something- something-v1 | 110K | 174 | - | - | - |
| [45] | SUN RGB-D | 47 scenes | 19 | - | - | - |
| [60]& [57] | UCF101 | 13320 | 101 | | 3 | RGB |
| [44] | UCF50 | 6676 | 50 | 25 | - | - |
| [17] | UoS-DB | - | 5 | - | - | - |
| [49] | UTD-MHAD | - | 27 | 8 | - | - |
| [55] | UTK Action3D | - | 6 | - | - | - |
| [48] | UT-Kinect | 199 | 10 | 10 | - | Skeleton/RGB |
| [27] | UTKinect- Action | - | 10 | 10 | - | - |
| [56] | VIRAT | 497 | 11 | - | - | - |
| [32], [39]& [28] | Weizmann | 92 | 10 | 9 | - | - |
| [12] | WVU | 517 | 11 | | 8 | RGB |



Figure 2.5 View of 8 cameras showing sample actions and sample actors

Computer vision research and applications, particularly through the incorporation of multiview and multimodal information, have been significantly enhanced by the exploration of human action recognition. Research in the area of human-computer interaction, intelligent video surveillance, and multimedia content understanding and management has allowed for the development of practical applications. These applications are being used in the real world, and have been made possible through these studies. To recognize actions from different viewpoints and modalities remains an active research area. And, availability of the video sequence database in huge, attracting researchers to do research in this field.

The year-wise comparative analysis based on research methodology in human activity recognition is present in Table 2.3. The literature survey has found that the neural network-based process is used the most instead of other methods.

Table 2.3 Year-wise comparative analysis of techniques

| Reference | Year | HMM | SVM | Directionality based algorithm | NN | DTW | Fusion | Fuzzy | PCA | Graph Base Model | Dictionary Learning | Sparse | DNN |
|-----------|------|-----|-----|--------------------------------|----|-----|--------|-------|-----|------------------|---------------------|--------|-----|
| [34] | 2002 | | | | | | | | | | | | |
| [21] | 2004 | ♦ | | | | | | | | | | | |
| [29] | 2006 | | | | | | | | | | | | |
| [16] | 2007 | | ♦ | | | | | | | | | | |
| [17] | 2008 | | | ♦ | | | | | | | | | |
| [30] | 2008 | ♦ | | | | | | | | | | | |
| [24] | 2008 | | | | ♦ | | | | | | | | |
| [31] | 2008 | | | | | ♦ | | | | | | | |
| [32] | 2008 | | | | | | ♦ | | | | | | |
| [26] | 2009 | | | | | | | ♦ | | | | | |
| [33] | 2009 | | | | ♦ | | | | | | | | |
| [22] | 2010 | ♦ | | | | | | | | | | | |
| [19] | 2011 | | | | ♦ | | | | | | | | |
| [39] | 2011 | | | | | | | | ♦ | | | | |
| [59] | 2012 | | ♦ | | | | | | | | | | |
| [35] | 2012 | | | | ♦ | | | | | | | | |
| [23] | 2012 | | | | ♦ | | | | | | | | |
| [36] | 2013 | | | | ♦ | | | | | | | | |

| | | | | | | | | | | | | | |
|------|------|--|---|--|---|--|--|---|---|---|---|---|---|
| [25] | 2013 | | | | | | | ♦ | | | | | |
| [28] | 2013 | | | | ♦ | | | | | | | | |
| [37] | 2015 | | | | ♦ | | | | | | | | |
| [11] | 2015 | | | | | | | | | ♦ | | | |
| [44] | 2015 | | | | ♦ | | | | | | | | |
| [12] | 2016 | | | | | | | | | | ♦ | | |
| [45] | 2016 | | | | | | | | | | | | ♦ |
| [46] | 2016 | | | | | | | | | | | | ♦ |
| [55] | 2016 | | ♦ | | | | | | | | | | |
| [27] | 2016 | | | | | | | | | | | ♦ | |
| [43] | 2016 | | | | | | | | | | | ♦ | |
| [38] | 2017 | | | | | | | | ♦ | | | | |
| [20] | 2017 | | | | ♦ | | | | | | | | |
| [60] | 2017 | | | | ♦ | | | | | | | | |
| [40] | 2017 | | | | ♦ | | | | | | | | |
| [58] | 2017 | | | | ♦ | | | | | | | | |
| [42] | 2018 | | | | | | | | | | | ♦ | |
| [47] | 2018 | | | | | | | | | | | | ♦ |
| [48] | 2018 | | | | | | | | | | | | ♦ |
| [15] | 2018 | | | | | | | | | | | | ♦ |
| [49] | 2019 | | | | | | | | | | | | ♦ |
| [41] | 2019 | | | | ♦ | | | | | | | | |
| [57] | 2020 | | | | | | | | | | | | |
| [56] | 2020 | | | | | | | | | | | | |
| [54] | 2021 | | | | ♦ | | | | | | | | |

Based on literature survey, The ROSE Lab was established collaboratively between Nanyang Technological University in Singapore and Peking University in China. Four data modalities are available i.e. Depth map, 3D connection information, RGB frames, IR sequences. In this work, the 3D joints and RGB information only uses. There are 40 distinct subjects between the age of 10 to 35 years. The dataset videos recording at three different angles, i.e., 45°, -45° and 0°, as shown in Figure 2.6. The dataset divide into three main categories: 40 activities regarding day-to-day activities, 9 about human health, and 11 that are mutual activities. This dataset varies in the number of subjects and ages of subjects [53].

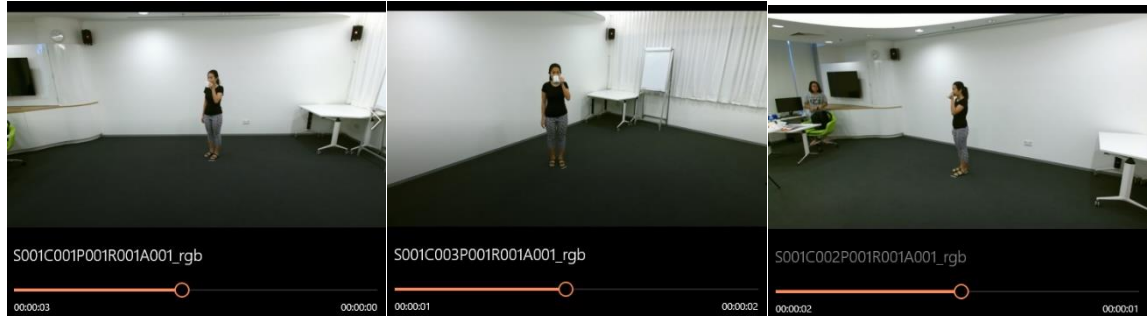


Figure 2.6 Three views in NTURGB+D (side view (+45°), front view (0°) and side view (-45°))

2.2 Research gap

The important points to fill the research gap for human activity recognition field is:

- 1) There are methods given by [39], [59], [35], [23], [11], [12], [48] have given high recognition rate according to their experimental setup. However, author gave method for handling a smaller number of action classes.
- 2) Multi-domain action recognition is consisting two things that is multiview and multimodal recognition. View and modality changes can lead to the presence of information from multiple domains, making action recognition more difficult [61]. Because RGB and depth has own characteristics such as illumination, backgrounds and appearance). Therefore, need a system which can handle different modalities.

At last, two main challenges in action recognition are how to discriminately represent multiview and multimodal action perceptual structure and how to learn classifiers to recognize classes with multi-domain data. According to the literature review, a state-of-the-art method for identifying actions will be proposed if data has viewed variation and a greater number of activities. The deep learning-based methodology is proposed in coming chapters 5 and 6 based on the survey presented in tables 2.1 and 2.3. The two datasets are used for the experimental approach that NTURGB+D and HMDB51 mentioned (Bold) in Table 2.2.

CHAPTER-3

SUBJECT SEGMENTATION OF ACTIONS FROM RGB FRAMES WITH SKELETON DATA

This chapter is about the techniques for segmentation. There are two ways for segmentation gives i.e, subject segmentation, which separates actions from video using the NTURGB+D dataset. Secondly, the experiment analysis on object segmentation using semantic segmentation using the CamVid dataset.

3.1 Subject Segmentation of RGB frames

The process of dividing a video into sections to identify actions within each frame is referred to as subject or video segmentation. It is required in various applications i.e. design CAD system capable of supporting visual effects in movies, understanding detailed scenes, creating virtual backgrounds, and automatically recognizing human behavior from video without background interference. A system has been developed to perform automatic activity segmentation in videos, which can adapt to the specific size of the input video. The data for experimentation is drawn from the NTURGB+D dataset. The methodology utilizes 3D skeletal information from RGB videos in the NTURGB+D dataset to achieve subject segmentation. Evaluation results were obtained by conducting experiments on five randomly selected action videos, showcasing performance and testing outcomes.

3.1.1 Literature Survey

Motion detection based on video images have number of steps such as preprocessing of video frames, segmentation to find the different objects, feature calculation, reduction of features, and classification. The system's accuracy is high if feature computation is performed on correctly segmented image/video frames. It uses arbitrary techniques or deep learning methods, which are the high edge methods. Nevertheless, the two main aspects of designing a deep learning model include the input and output sets for training or testing the network and tuning the hyperparameters. The quality of the input information plays a major role in determining the precision of the model. Therefore, segmenting video frames is a significant step in designing the models. The work is based on action segmentation to isolate the object movement and the background using trajectories for classification [62]. Segmentation of the frame based on unsupervised learning has been proposed by [63]. The system based on bag of visual word features to segment video actions

provided by [64]. [65] an approach is given to define actions and features useful for neighborhood segmentation. The data-intensive techniques [66] can help to design a automatic subject segmentation based in 3D joint data. Action detection based on action localization is useful when background interference is not considered [67].

Segmentation of temporal actions is becoming increasingly common when using approaches that train against timestamp annotations [68]. Researchers have developed various structures that can separate the object movement and the background from video frames. The Global2Local structure is an approach that can use to segment video actions. This approach can find the action based on motion given by [69]. Segmented video use in dynamic manifold warping to calculate similarity of the motion [70]. In this work, a segmentation method using 3D skeletal information is specified. Other modalities, such as depth maps can also be used for the subject segmentation. You can improve action detection rate by using depth map [71]. It is possible to use the energy of motion history images to segment long sequence videos [72]. Segmented action videos helps to improve the action recognition systems, one such type of system automatically based on pose streams in the form of joints [73]. Contour-based segmentation is another method to compute features of still images [74].

To date, few proposed approaches that can be used for behavioral segmentation have been discussed. When feeding input videos to deep learning models or machine learning models, the system does not focus on the action but also considers background details. Segmented video frames can be fed to systems that focus on training actions instead of whole frames. In this work, we demonstrated an approach that can be used for segmenting measures on the NTURGB+D dataset. This segmentation approach allows us to design more efficient systems.

3.1.2 About Dataset NTURGB+D

The development of depth sensors incorporated in cameras has enabled us to acquire various forms of data. These four modalities include RGB video, depth map, IR sequence, and 3D structural information, all of which have their benefits and can be applied in action recognition. Two of the four types have been utilized to perform subject segmentation: RGB and 3D structural data. The NTURGB+D dataset was created to support data-intensive techniques such as deep learning approaches. This dataset is composed of 56880 videos of resolution 1920X1080 taken from 40 subjects and the corresponding 3D joint information with 25 body joints shown in Figure 3.2. The NTURGB+D dataset thus provides a useful platform for building accurate human activity detection models [53].

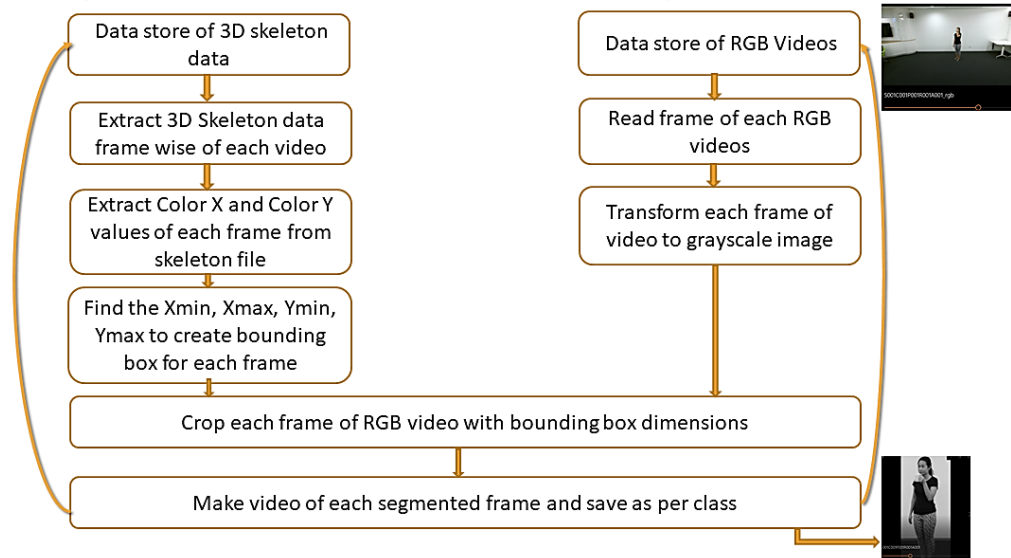


Figure 3.1 Steps for subject segmentation

3.1.3 Steps for subject segmentation

The following portion explores the technique utilized for identifying actions. It explains the circumstances under which this algorithm can be utilized in platform such as MATLAB. The task is separated into several distinct components as depicted in Figure 3.1. This approach has been developed with an emphasis on deep learning techniques. The initial move is to make a data repository that the deep learning algorithm can use for training, validation, and testing purposes. In the process, just one data set called NTURGB+D containing 3D and RGB video is used. Next, the 3D data must be unpacked and a bounding box with [1x4] dimensions must be created as shown in Figure 3.1. A bounding measure based on the characteristics of the initial frame of 3D data. Subsequently, each video frame is converted to grayscale. Subsequently, each video frame is trimmed according to the bounding box size. In conclusion, a video consisting of segmented frames is generated and classified for storage. This information can then be utilized for deep learning methods.

3.1.4 Experimental Analysis

A 3D skeleton can be incredibly helpful in revealing changes in a person's posture over time. The RGB video was divided into segments using the 3D skeletal data. This type of skeleton usually contains a series of 3D coordinates for different body joints. Segmentation refers to the division of a digital image into separate sections (also known as image objects). The aim of dividing an image/frame is usually to enhance its comprehensibility and simplify analysis. Several steps are involved in obtaining results, including:

- *Construct a repository for three-dimensional skeletal information and colored video.*

The first step is to establish data repositories for the 3D skeletal and RGB videos. This step is crucial for organizing the segmented videos into classified folders. This is a helpful feature when deep learning models need to identify activity.

- *Retrieving Color X and Color Y from the 3D Skeletal Data*

The NTURGB+D dataset consists of 3D skeleton data connected to each video, indicated by a skeleton file extension. This data contains 25 skeletal joints, thumb tracking, hand edge tracking, opening and closing hand gestures, as well as other significant aspects of human body movements. Such information is essential for motion recognition, as 25 pieces of information are required for each video frame. Each video's skeleton file extracts 25 pieces of information across all platforms. Color X and Y extraction are performed for each video frame to achieve subject segmentation. As shown in Figure 3.2, the steps to get the desired output are outlined.

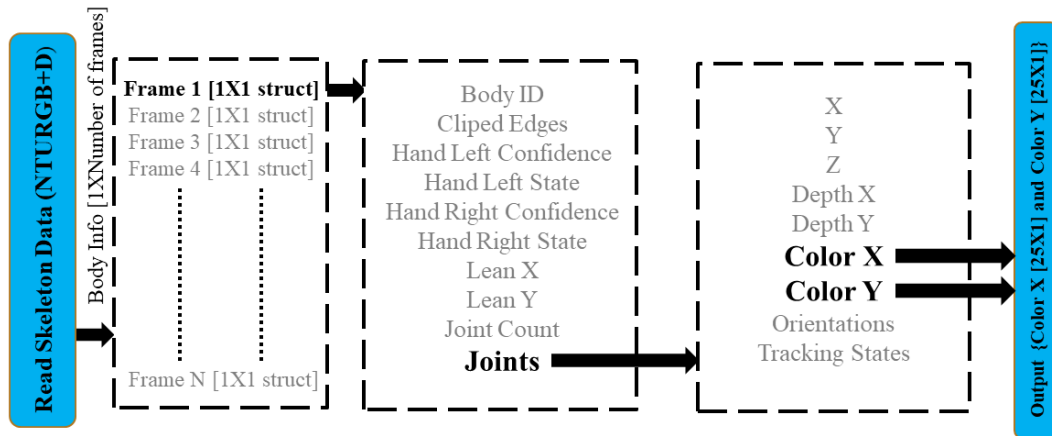


Figure 3.2 Data of Joints Color X and Joints Color Y

- *Subject segmentation bounding box dimensions*

The outputs designated as Color X and Color Y from each frame serve as inputs for creating the video's boundary box. Xmin, Xmax, Ymin, and Ymax are the four dimensions of the boundary box and can be calculated using the minimum and maximum values of Color X and Color Y. The video frame is then trimmed to match the dimensions of the boundary box as stated in Equation 3.1. Figure 3.3 outlines the steps to find the boundary box dimensions and trim to find the action-specific part.

$$Bbox = [Xmin, Ymin, (Xmax - Xmin), (Ymax - Ymin)] \quad (3.1)$$

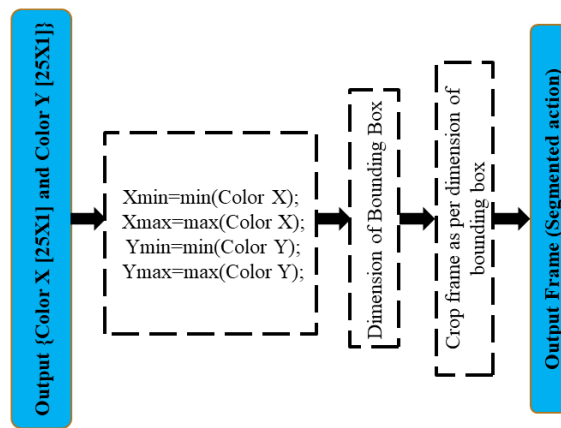


Figure 3.3 Steps for subject segmentation per frame

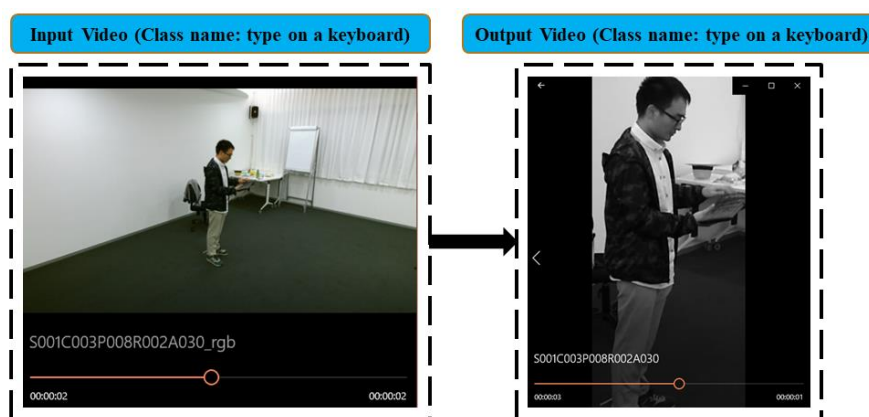


Figure 3.4 Results of the experiment on the "type on a keyboard" class were shown with an input video (on the left) and a video segmented by action (on the right)



Figure 3.5 The outcome of the experiment on the "salute" class [illustrated by the input video (on the left) and the action-segmented video (on the right)]

- *Subject segmentation of RGB video*

The NTURGB+D data set contains 60 action labels with 56880 videos. Grades A001 to A049 feature daily activities and health conditions for one subject, while grades A050 to A060

include joint activities for two. Five random videos were selected from everyday actions and joint activities, featuring keyboard typing, saluting, pushing, clapping, and drinking. The segmented videos can be seen in Figures 3.4 to 3.8.

In a video, there is a background and subject. The frame size of RGB from NTURGB+D is 1980x1080. To segment each frame of RGB video, skeleton 3D information uses and segments a video with the dimension of a bounding box given in equation 3.1. After segmentation, the frame size is reduced by more than half. The size of the RGB video frame and segmented video frame are given in table 3.1.

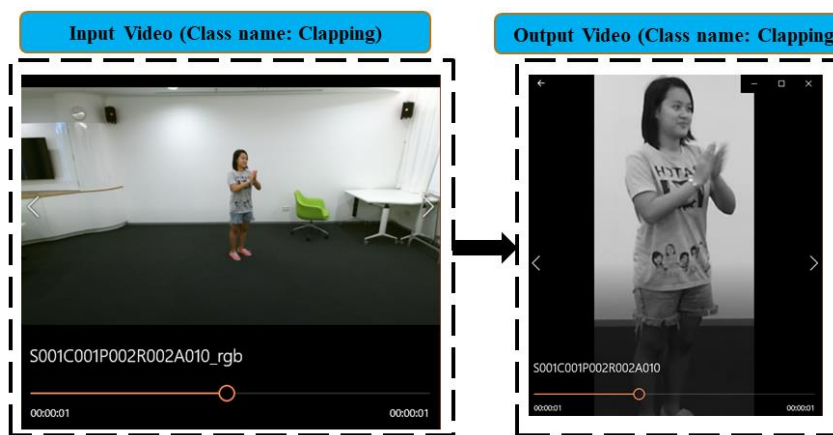


Figure 3.6 The results obtained from the experiment on the "clapping" class [with an input video (on the left) and a segmented action video (on the right)]

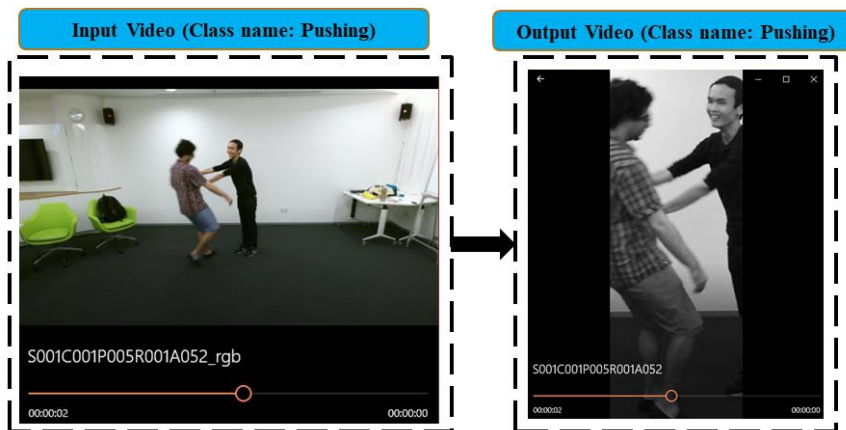


Figure 3.7 The outcome of the experiment on the "pushing" class [with the input video (on the left) and the video of the segmented action (on the right)]

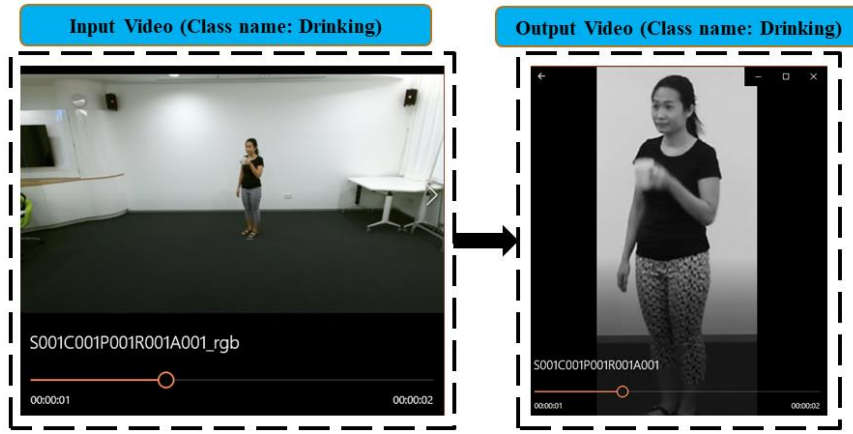


Figure 3.8 The results of the experiment conducted on the class "drinking" were depicted in the form of an input video on the left and an action-segmented video on the right.

Table 3.1 Size of video and segmented video of NTURGB+D dataset

| Detail of Video | | Video (Size) | | | Segmented Video (Size) | | | Segmentation percentage |
|--------------------|-------------------------|--------------|--------|----------------|------------------------|--------|----------------|-------------------------|
| Name of Activity | Video Label in NTURGB+D | Width | Height | Total (Pixels) | Width | Height | Total (Pixels) | |
| type on a keyboard | S001C003P008R002A030 | 1920 | 1080 | 2073600 | 253 | 596 | 150788 | 92.73 |
| salute | S001C001P008R002A038 | 1920 | 1080 | 2073600 | 229 | 542 | 124118 | 94.01 |
| clapping | S001C001P002R002A010 | 1920 | 1080 | 2073600 | 217 | 511 | 110887 | 94.65 |
| pushing | S001C001P005R001A052 | 1920 | 1080 | 2073600 | 217 | 627 | 136059 | 93.44 |
| drinking | S001C001P001R001A001 | 1920 | 1080 | 2073600 | 198 | 473 | 93654 | 95.48 |

3.2 Semantic Segmentation for Object Recognition

Object detection and recognition from images or videos is a critical task for determining the presence of an entity in them. Analysis of recognition and identification has become crucial for security, monitoring, personal storage or self-driving applications, and so on. This project introduces a system for scene detection. A Deep Lab v3+ system with Resnet18 initialized weights has been utilized in this approach. The CamVid database has been put to use to detect various scenes from video frames. It consists of 701 total RGB frames, which have pixel-wise segmentation labeling and RGB values label. To evaluate the model's performance, three parameters were utilized, namely, accuracy, IoU and score. Upon analysing these parameters, it was determined that the model is well-suited for recognizing scenes from video frames.

3.2.1 Literature Survey

Computer vision is a multi-disciplinary field that enables the automated analysis, extraction and understanding of information from individual images or collections. Identifying scenes from images and videos is a highly sought-after task in computer vision and is applied across a range of

fields, from industry to consumer and general purposes. Segmentation is a particularly hard task to complete under pre-processing techniques. The advantage of vision or mass perception is a significant computer vision challenge applied in various fields, such as autonomous vehicles. [75], human-machine interaction, computer graphics [76], traditional search engines and augmented reality.

Semantic segmentation is capable of handling the deep learning architecture used for categorization. Locating items in images and videos is the initial step in classification. This provides us with the class and other data that aids in recognizing objects with their spatial information. This aims to assign a tag to each pixel in a photograph or video [77]. Pixel-based classification can be a valuable tool in a range of scenarios. It is a powerful technique that can be applied to various purposes [75][76][77].

Segmenting an image or video is crucial in image or video processing. Semantic segmentation is utilized in the initial stage of computer vision tasks. Another existing technique relies on databases. A strategy that uses orientated patterns, colors, and textures is proposed in this instance [78]. It is possible to train a CNN network end-to-end for multiple groups with joint segmentation to maintain each group's foreground. This can be achieved through the use of two sub-networks, one for region extraction and the other for segmentation [79]. Semantic segmentation is also employed in hash-based multimedia search due to its efficiency and cost-effectiveness [80]. The majority of the cutting-edge approaches require extensive computing resources, and they can be employed in industrial settings, such as to comprehend a scene [81]. Semantic segmentation can also be used for city scenes. Gives good enough results when used with poorly labeled data [82]. Automatically digitize maps for scientific and industrial use instead of manually interpreting scenes using convolutional neural networks [83]. A new graphical approach-based model is suggested to improve existing procedural methods. This model takes information from multiple functions and then compares the performance of those functions with other methods in terms of execution time [84]. Segmentation includes both object segmentation and pixel segmentation. SPCO (Seed-Picking Crossover Optimization) is proposed as an approach to address the requirement for both high- and low-level features in order to accomplish pixel-by-pixel classification and enhance performance. This approach utilizes a Conditional Random Field (CRF) trained model to select appropriate features and accurately classify objects [85]. Developing a neural network that uses the newly created Spatio-temporal continuous (STC) semantic segmentation for video has been achieved [86]. This proposed technique is used to segment video imagery. Semantic segmentation plays an important role in real-world scenarios like the robotics domain. When robots need to make decisions based on intricate pictures, semantic segmentation

can be used to upgrade the system. This facilitates making intricate visual choices [87]. To enhance the segmentation outcome, FCN sequential maps can be combined to execute image classification tasks [88]. Designing a system that efficiently detects prohibited items [89]. A network can be constructed by integrating two sub-networks that guide focus towards a specific task [90].

Analyzing objects in videos or images is essential, which can be achieved through deep learning networks. This method of object detection is a crucial part of preprocessing for video or image data. By employing this process, one can determine whether the desired objects are present in the video. This work conserved the network used for object identification from video frames.

3.2.2 Steps for Object recognition using semantic segmentation

- *Image Pre-processing*

For this particular deep neural network, the size of the input layer determined the dimension of the image. For the research at hand, the image input dimension was set as 720x960 pixels with three color channels.

- *The CNN for segmentation task*

Image segmentation can pose difficulties when viewed on a distant computer. To tackle this challenge, various algorithms have been created to enhance outcomes. These algorithms include the watershed method, image thresholding, k-means clustering, and graph partitioning, but the most successful results have been achieved using deep neural networks. These networks comprise several layers, such as convolutional, activation, batch normalization, and max/min pooling layers, which assist in recognizing the spatial elements of the input image. Combining convolutional and pattern layers in an encoder-decoder structure is necessary to attain the desired output, which optimizes weightings with extensive detail. The layer below the input is an element of the encoder, while the layers responsible for generating the samples form part of the decoder [91]. In this investigation, the DeepLab v3+ network was utilized as a convolutional neural network for semantic image segmentation. The network employs an encoder-decoder structure and seamless integration, eliminating the requirement of connecting to image segments. A depiction of the encoder and decoder is shown in Figure 3.9.

When training a model for semantic segmentation, the encoder produces a tensor that includes information regarding the objects like shape and size. The decoder then uses this data to make a segmentation map [92].

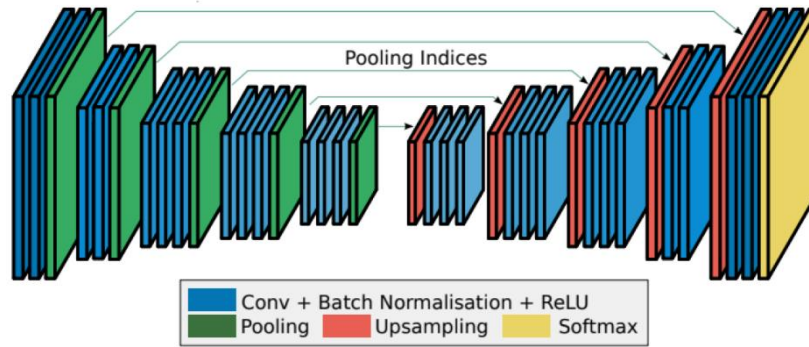


Figure 3.9 Encoder-Decoder Structure [91]

3.2.3 About Dataset CamVid

This research utilized the Cambridge Driving Labeled Video Database (CamVid) as a video-based object analysis database. The video footage was captured using an onboard high-resolution 3CCD Panasonic HVX200 digital camera, capable of recording 960x720 pixels at 30 frames per second. Four HD video streams, titled 0001TP (8:16), 0006R0 (3:59), 0016E5 (6:19), and Seq05VD (3:40) for a total duration of 22:14, were obtained from the CamVid website. These videos encompass a diverse range of classes, including cars, pedestrians, and cyclists. A total of 32 classes were identified from the video sequences [93]. In this instance, we use shades of red, green, and blue to illustrate the spatial data, as shown in Figure 3.11.

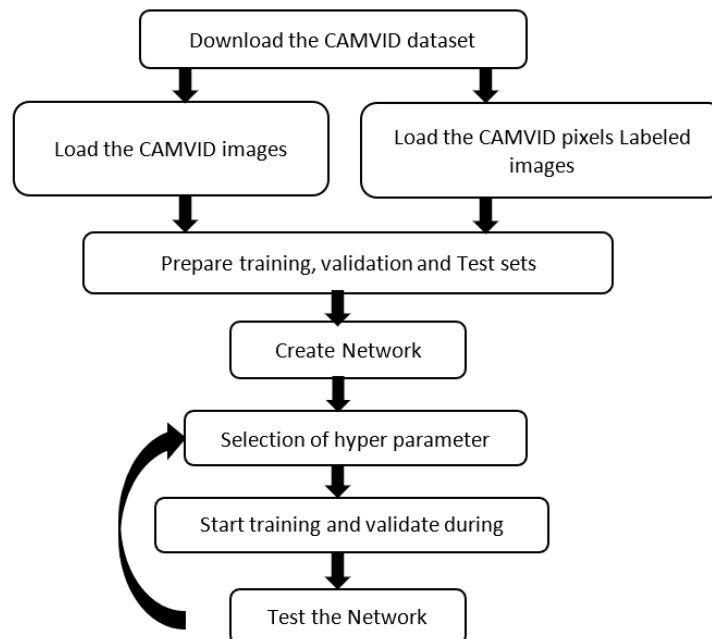


Figure 3.10 Steps to test network

| | | | | |
|-----------------|--------------------|--------------|----------------|-----------------|
| Void | Building | Wall | Tree | VegetationMisc |
| Fence | Sidewalk | ParkingBlock | Column_Pole | TrafficCone |
| Bridge | SignSymbol | Misc_Text | TrafficLight | Sky |
| Tunnel | Archway | Road | RoadShoulder | LaneMkgsDriv |
| LaneMkgsNonDriv | Animal | Pedestrian | Child | CartLuggagePran |
| Bicyclist | MotorecycleScooter | Car | SUVPickupTruck | Truck_Bus |
| Train | OtherMoving | | | |

Figure 3.11 The following is a list of 32 object class names, along with the color used to label [93]

Per-pixel labeling with RGB values gives you an accurate look and shape experience. Examples of video frames with corresponding images labeled pixel by pixel, as shown in Figures 3.12 (a) and (b), respectively. Figure 3.13 shows the length of the video sequence. A specified number of marked frames accompanied the corresponding video length for each video. The total length of all videos is 10 minutes. Looking at each frame of the video, several objects can be labeled, as shown in Figure 3.11 [93].

The CamVid dataset is useful for researchers looking to create algorithms and networks that can identify objects through semantic segmentation. This database can be employed for a range of applications. This algorithm can handle object detection, pedestrian detection, and object tag propagation [93].



Figure 3.12 (a) A sample of a frame taken from the CamVid dataset, (b) An example of a frame labeled on a pixel-by-pixel basis from the CamVid dataset.

3.2.4 Experimental Analysis

In this study, a Deeplabv3+ deep learning network was created in order to carry out semantic segmentation, which involves assigning each pixel in an image/video to a specific category. The network was created using pretrained weights from a Resnet-18 network, a

Convolutional Network, also known as a Deep Architecture Network [94]. Deeplabv3+ is a Convolutional Neural Network (CNN) form that utilizes millions of pictures from the ImageNet database to train. With its advanced training, it can recognize over a thousand distinct categories or classes. By leveraging transfer learning concepts, deep architectures can be used to classify new classes. Deeplabv3+ is used as a CNN specifically designed for semantic image segmentation to execute the training.

| labeled sequence name | original sequence name | frame rate in fps | number of labeled frames | corresponding duration |
|-----------------------|------------------------|-------------------|--------------------------|------------------------|
| 0001TP_L | 0001TP | 1 | 124 | 2:04 |
| 0016E5_1Hz_L | 0016E5 | 1 | 204 | 3:24 |
| 0016E5_15Hz_L | 0016E5 | 15 | 101 | 0:06 |
| 0006R0_L | 0006R0 | 1 | 101 | 1:41 |
| Seq05VD_L | Seq05VD | 1 | 171 | 2:51 |
| | Total | | 701 | 10:06 |

Figure 3.13 Videos with a title, frames per second, the quantity of frames, and length of time are labeled [93]

Completing a neural network is the foremost and most critical step when it comes to object recognition using semantic segmentation. After that, additional steps must be taken to accomplish the task. Once Resnet18 is complete, install the relevant network into your programming software. Suitable for applications that require limited processing resources. Once a dataset of CamVid images and corresponding CamVid pixel labels has been loaded, it is important to perform a split of the data into training, testing, and validation datasets. Pre-processing the datasets is essential for ensuring the efficient performance of deep neural networks. An effective method of examining data is to establish a DeepLabv3+ network built on the ResNet18 architecture. Finally, determining the most appropriate network for a particular use case involves evaluating and adjusting the relevant hyperparameters through practical testing [95]. Networks were always analyzed after several rounds of training, validation, and testing on data. The number of depths or total number used in the network increases the value of the network to achieve a higher detection rate. This involves various hyperparameters called training options.

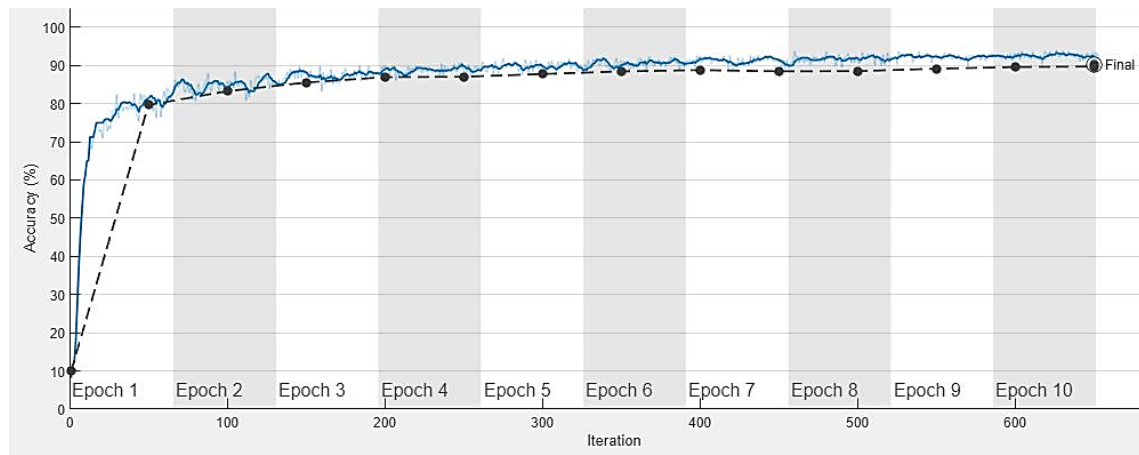


Figure 3.14 Plot showing the development of accuracy over time as a result of the training process (iterations)

Figure 3.14 shows the training progress chart, showing how the accuracy changes exactly from epoch to epoch. The solid blue line shows how the training of data for each epoch. Also, the black coloured line (dotted) shows how validation tests are handled. This algorithm was used to train stochastic pulse rate gradient descent. Various hyperparameters were used to tune the network. A learning rate of 0.0003 is used and decreases with each epoch. These will help you network and find the best solution. The network was tested on both validation data and individual test data. Looking at Figure 3.11, the initial accuracy is 10%, but the accuracy reached 80% during training only in the first epoch. The training history graph shows a decline in the ultimate validation accuracy, which hit 90.2808% following the 10th epoch. In general, this type of network may be preferred. Once the network is trained, the training process will be terminated when the validation accuracy reaches a steady rate.

There are three situations that can arise when working with deep neural networks. The network may overfit, underfit, or align identically with both the training and validation datasets. In order to make sure that the network is functioning correctly, we need to review the training and validation losses for each iteration. Figure 3.15 supplies data regarding the transformation in the losses in successive iterations. A higher training loss than the validation loss suggests that the network may suffer from underfitting.

An overfitting issue in the network can be indicated if the validation loss is larger than the training loss. On the other hand, if the validation loss and training loss are approximately equal, then the network is more likely to perform well. As seen in Figure 3.11, the training loss (represented by the solid red line) is almost 2.5 at the start of the first few iterations and then decreases gradually. Similarly, the validation loss (represented by the black dashed line) in Figure 3.12 is nearly equal to the training loss and decreases following epoch 2. The validation loss

approached 0.4224 at the end of epoch 10, near the 600th iteration mark.

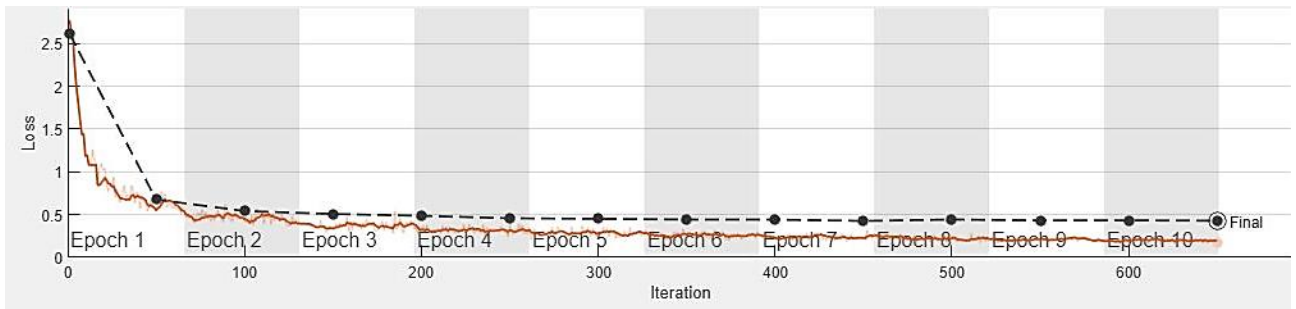


Figure 3.15 Plot of loss over the number of iterations during training

Table 3.2 Recognition rate per classes of testing data

| Observation class Vs Target class | Sky | Building | Pole | Road | Pavement | Tree | SignSymbol | Fence | Car | Pedestrian | Bicyclist |
|-----------------------------------|--------|----------|--------|--------|----------|--------|------------|--------|--------|------------|-----------|
| Sky | 0.9505 | 0.0043 | 0.0085 | 0 | 0 | 0.0352 | 0.0011 | 0 | 0.0004 | 0 | 0 |
| Building | 0.0086 | 0.843 | 0.0477 | 0.0002 | 0.0125 | 0.0219 | 0.0355 | 0.0047 | 0.01 | 0.0137 | 0.0021 |
| Pole | 0.0191 | 0.0891 | 0.7171 | 0.0017 | 0.0166 | 0.0542 | 0.0538 | 0.0086 | 0.0137 | 0.024 | 0.0023 |
| Road | 0 | 0.0002 | 0.0015 | 0.9532 | 0.0319 | 0.0003 | 0.0001 | 0 | 0.0093 | 0.001 | 0.0025 |
| Pavement | 0 | 0.0063 | 0.015 | 0.076 | 0.881 | 0.0014 | 0.0005 | 0.0019 | 0.0053 | 0.0093 | 0.0033 |
| Tree | 0.0435 | 0.0295 | 0.0328 | 0.0007 | 0.0017 | 0.8619 | 0.0101 | 0.0111 | 0.0045 | 0.0035 | 0.0007 |
| SignSymbol | 0.0008 | 0.1035 | 0.0433 | 0.001 | 0.0042 | 0.0483 | 0.7536 | 0.0038 | 0.0312 | 0.0097 | 0.0005 |
| Fence | 0 | 0.0588 | 0.0676 | 0.0005 | 0.0176 | 0.0412 | 0.0037 | 0.7644 | 0.0171 | 0.0277 | 0.0013 |
| Car | 0.0012 | 0.0254 | 0.0172 | 0.0098 | 0.0053 | 0.0044 | 0.0074 | 0.0054 | 0.8989 | 0.02 | 0.0049 |
| Pedestrian | 0 | 0.0455 | 0.0506 | 0.0016 | 0.0104 | 0.0041 | 0.0045 | 0.0053 | 0.0186 | 0.8445 | 0.0151 |
| Bicyclist | 0 | 0.0131 | 0.0048 | 0.008 | 0.0059 | 0.0078 | 0.0031 | 0.0002 | 0.0129 | 0.0792 | 0.865 |

The recognition rate is crucial to comprehend, as it shows how successfully our system authenticates the test data. Table 3.2 presents the separate recognition rates of 11 classes evaluated using only the test data. Both RGB and pixel label data have 140 frames. The first column in Table 3.2 represents the predicted labels, and the first row shows the target class names. For example, the frames in the test data were correctly identified as "sky" 95.05% of the time. Other examples include 84.3% as "building", 71.71% as "pole", 95.32% as "road", 88.1% as "pavement", 86.19% as "tree", 75.36% as "signsymbol", 76.44% as "fence", 89.89% as "car", 84.45% as "pedestrian", and 86.5% as "bicyclist". The green box highlights the highest accuracy for each target class. Table 3.2 also demonstrates the accuracy variation.

Figures 3.16 (a) and (c) depict the RGB frames from the CamVid dataset, while semantically segmented frames can be seen in Figures 3.16 (b) and (d). Upon examination of the semantically segmented frames, it becomes apparent that the semantic segmentation performed

well in identifying the road, sky, and buildings, but was not as effective in recognizing pedestrians and other objects. Semantic segmentation's effectiveness depends on the area's complexity, as it is more successful in simpler parts of the framework. Semantic segmentation results are less satisfactory when objects are small and complex.

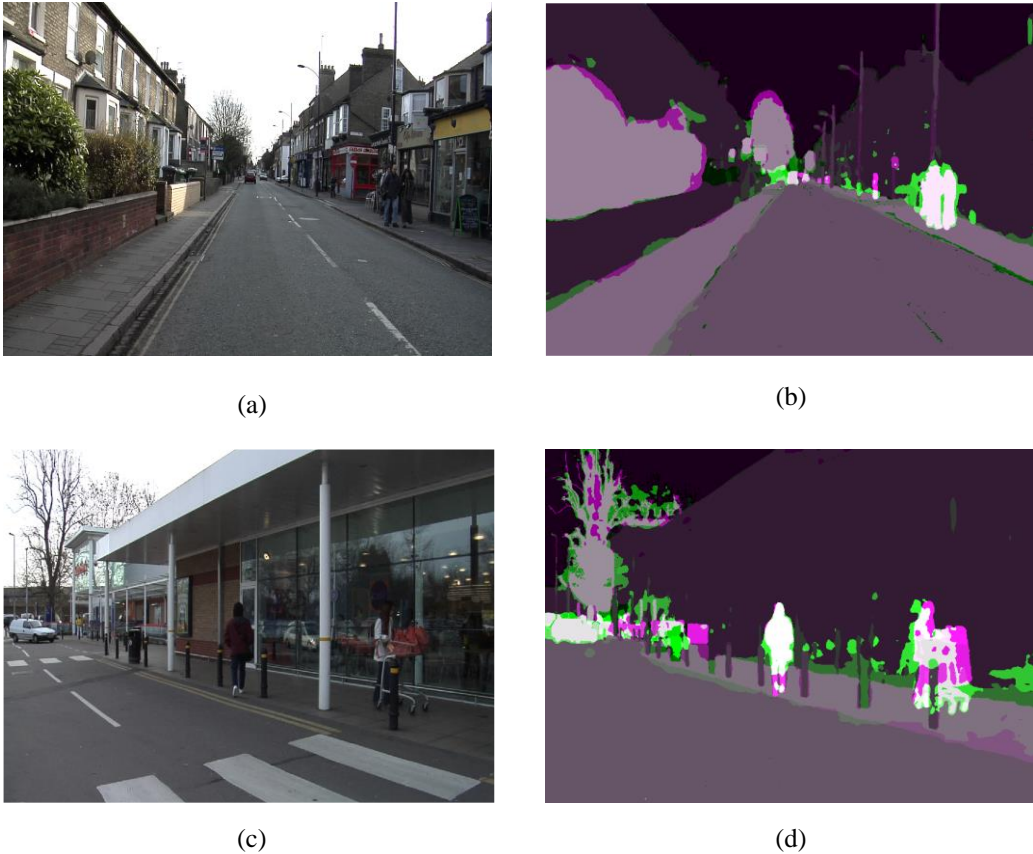


Figure 3.16 The evaluation of trial data, where (a) and (c) are RGB frames from the CamVid dataset, (b) and (d) are the corresponding segmented images, is presented.

Constructing the optimal model for all data and weights through functions can pose a significant challenge. It is important to be able to evaluate models with different parameters in order to determine the overall model. To do this, three metrics can be used: accuracy, Intersection of Union (IoU), and ratings. Accuracy measures the correspondence between the predicted class and the target class, while IoU (Intersection over Union) is a value between 0 and 1 that reflects the overlap between the predicted box and the actual bounding truth. A higher IoU value indicates that the boxes overlap more, while a lower value indicates that the boxes have more space between them. Ratings are indicators of the class to which an observation belongs, with the maximum points per class listed in Table 3.3. Table 3.3 reveals that the highest precision is achieved for the class 'Sky' and the lowest precision is for the class 'Pole'. All the metrics referred to can be found in

Table 3.3.

Table 3.3 Class wise accuracy, IoU and Mean scores values

| Class Parameters | Accuracy | IoU | Mean Scores |
|-------------------|----------|--------|-------------|
| Sky | 0.9505 | 0.9088 | 0.9044 |
| Building | 0.843 | 0.8118 | 0.6711 |
| Pole | 0.7171 | 0.2629 | 0.6034 |
| Road | 0.9532 | 0.9337 | 0.8293 |
| Pavement | 0.881 | 0.7411 | 0.7551 |
| Tree | 0.8619 | 0.773 | 0.7234 |
| Signsymbol | 0.7536 | 0.4112 | 0.5286 |
| Fence | 0.7644 | 0.5932 | 0.5398 |
| Car | 0.8989 | 0.791 | 0.7543 |
| Pedestrian | 0.8445 | 0.4288 | 0.596 |
| Bicyclist | 0.865 | 0.6222 | 0.5634 |

3.3 Summary

This chapter is related to segmentation techniques, from which first is subject segmentation using Skeleton 3D information of NTURGB+D dataset and second, object segmentation using pixel-wise labeled technique which is called as semantic segmentation using CamVid dataset.

The work related to subject segmentation is an approach to segmenting all classes of RGB video using skeletal 3D information. This dataset boasts a substantial size, making it ideal for applications requiring many classes or perspectives. The size of the segmentation window is determined based on the RGB video using skeletal 3D information, making this technique practical for all NTURGB+D videos. The Experimental Results section showcases some randomly selected video results. Visualizing the output of the segmented video reveals generally good performance, except for class where includes mutual actions. If these categories are to be considered, the window may need to be modified accordingly. Also, the result shows that based on the segmented percentage from which it has been observed, the maximum segmented rate is 95.48%. The segmented videos can be used for any experiment. Future work will improve this technique when multiple people are in the video.

Secondly, The work is related to utilizing a deep neural network for semantic segmentation-based object recognition and scene classification into 11 classes using the CamVid dataset. The DeepLab v3+ model with resnet18 was trained and tested with this dataset. The performance was evaluated based on three metrics: accuracy, IoU, and score, with the maximum precision for each class reported in the experimental results. The accuracy of class where an area is more in the image like "Sky" is 0.9505, which is the maximum. Wherever the area is small, like classes named "pole", "SignSymbol" and "Fence", the accuracy is less. Despite difficulties

recognizing smaller objects, the model performed well for larger objects. For IoU, it is minimal for the class "Pole" i.e. 0.2629, due to overlaps between the predicted and actual box and maximum for class "Sky" i.e. 0.9088. The mean scores indicate how many instances it classifies correctly, so the maximum score is for class "Sky" i.e. 0.9044, and the minimum for "SignSymbol" is 0.5286. Semantic segmentation has many potential uses in self-driving cars, behavior identification, and healthcare. In order to be successful, the initial step is to pinpoint available objects. Future work can focus on improving the recognition of smaller objects in images and videos, and if successful, this deep neural network model could be suitable for semantic segmentation.

CHAPTER-4

FEATURE EXTRACTION FOR OBJECT DETECTION

For recognizing actions, the first step is feature extraction. This chapter is about feature extraction for object detection. The histogram optical flow used to give information about how object movement from one frame to another frame and second is histogram of oriented gradient which is used to find the presence of object in an image. The both explained in next sections.

4.1 HOF (Histogram optical flow) features

Motion estimating is one of the methods which determines the movement from one frame to another in the videos. For an application of action recognition, choosing the optical flow can be an essential feature for recognizing actions. The optical flow consists of the information of the moving subject and objects in the video frames. This work analyzes four motion-estimating visual flow methods (Farneback, Horn Schunck, Lucas Kanade, and Lucas-Kanade Derivative of Gaussian explored based on visualization and PSNR. The NTURGB+D dataset uses for the analysis of experimental results.

Recognizing human actions is currently a formidable challenge. Various methods have been proposed in [96], which can use for pre-processing and recognizing activities. Motion estimation features are one of the main steps required to know activities in video sequences. The optical estimation technique is in [97]. The optical flow gives motion information and helps to find an interesting point used to recognize actions.

There are various non-parametric methods in which motion descriptor calculation helps estimate flow between one and the next frame. The flow descriptors aggregate the histograms on the temporal axis. The gradient-based methodology improves action recognition tasks [98]. Various other handcrafted features like SIFT, HOG, GIST, and MHI [99]. These features have more focus on spatial information but not that much on temporal characteristics. The human motion analysis features the identification of human body shape, the relation of motion from one frame to the next, and one aspect of human activity recognition [100]. The optical flow based on warping explores provided high accuracy [101]. This method can reduce the angle errors while computing optical flow. The optical flow can be used for video object segmentation, as mentioned in [102]. This model is attention based, which can use for object detection. Human activity recognition for video surveillance can design using an optical flow feature. Optical flow features include information related to the movement of subjects [103].

4.1.1 Methods for HOF

To calculate optical flow, techniques are analyzed in this work. The methods are Farneback [104], Horn Schunck [105], Lucas Kanade [106], and Lucas-Kanade Derivative of Gaussian [106].

The HOF features are widely used for optical flow estimation by differential equations. Despite calculated differences, the measurement can include three stages of processing. First, pre-filtering to extract interest points. Second, compute the spatial-temporal derivative (called velocity vectors). Third is integrating measurements to produce flow fields [107]. The techniques employed to determine the object's velocity and direction of movement from frame to frame.

For the computation of the optical flow features, can use the equation 4.1 as follows:

$$I_x u + I_y v + I_t = 0 \quad (4.1)$$

In equation 4.1, I_x , I_y and I_t are the brightness derivatives.

- *Algorithm 1: Horn Schunck* [104][107]

The method estimates a velocity vector by considering the flow features are smooth across the whole image.

$$E = \iint (I_x u + I_y v + I_t)^2 dx dy + \alpha \iint \left\{ \left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial u}{\partial y} \right)^2 + \left(\frac{\partial v}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 \right\} dx dy \quad (4.2)$$

In the above equation 4.1 [104], the spatial derivative of optical velocity components is $\frac{\partial u}{\partial x}$ and $\frac{\partial u}{\partial y}$, α is the scaling factor that is related to smoothness. The method used to minimize equation 4.2 for each pixel in the frame with equations 4.3 & 4.4 [104] is as follows:

$$u_{x,y}^{k+1} = u_{x,y}^{-k} - \frac{I_x [I_x u_{x,y}^{-k} + I_y v_{x,y}^{-k} + I_t]}{\alpha^2 + I_x^2 + I_y^2} \quad (4.3)$$

$$v_{x,y}^{k+1} = v_{x,y}^{-k} - \frac{I_y [I_x u_{x,y}^{-k} + I_y v_{x,y}^{-k} + I_t]}{\alpha^2 + I_x^2 + I_y^2} \quad (4.4)$$

The velocity estimator for pixel (x, y) is $[u_{x,y}^k \text{ and } v_{x,y}^k]$ and neighborhood average of $[u_{x,y}^k \text{ and } v_{x,y}^k]$ is $[u_{x,y}^{-k} \text{ and } v_{x,y}^{-k}]$. The initial velocity is zero. The steps to solve velocity vectors are as steps:

Step 1: Figuring of the brightness derivative i.e. I_x and I_y with Sobel kernels and its transposed form for each pixel in the first image.

Step 2: Computation of I_t , i.e. difference in first and second image using $[-1 \ 1]$ kernel which basically used to determine the difference or change.

Step 3: Computation of the velocity average of each image pixel using $[0 \ 1 \ 0; 1 \ 0 \ 1; 0 \ 1 \ 0]$ as convolution kernel.

Step 4: Iteratively solving of u and v .

- *Algorithm 2: Lucas Kanade* [104][107]

To solve horizontal u and vertical v optical flow features, the Lucas Kanade divides one frame into parts, assuming that each region has constant velocity. The method used weighted least square fit by minimizing the equation 4.5 [104] as follows:

$$\sum_{x \in \phi} W^2 [I_x u + I_y v + I_t]^2 \quad (4.5)$$

In equation 4.6 [104], “W” is the window function that focuses the constraints at each section's centre. The final solution to minimizing the part is

$$\begin{bmatrix} \sum W^2 I_x^2 & \sum W^2 I_x I_y \\ \sum W^2 I_x I_y & \sum W^2 I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} W^2 I_x I_t \\ W^2 I_y I_t \end{bmatrix} \quad (4.6)$$

Step 1: Figuring the I_x and I_y by using the kernel $[-1 \ 8 \ 0 \ -8 \ 1]/12$ with its transposed form.

Step 2: Computation of I_t , i.e, the difference of images 1 and 2 using the $[-1 \ 1]$ kernel.

Step 3: After, step 2 required to smooth the gradient components I_x , I_y and I_t by using 1D kernel.

Step 4: By solving of the 2-by-2 linear equations for each pixel and compare the eigenvalues with the threshold to calculate the u and v .

- *Algorithm 3: Lucas-Kanade Derivative of Gaussian* [107]

The Lucas Kanade method computes I_t using the Gaussian filter. For solving the Optical flow constraint equation for u and v .

Step 1: For computing I_x and I_y step follows as below:

- Perform temporal filtering by using the Gaussian filter.
- Perform spatial filtering by using the gaussian filter's derivative for smoothing each frame.

Step 2: Computation of the difference, I_t between images 1 and 2 using the $[-1 \ 1]$ kernel

- Perform temporal filtering by using the derivative of the Gaussian filter.
- Perform spatial filtering by using filter Step 1 (b) on the output of temporal filtering.

Step 3: Using a gradient smoothing filter, smooth gradient components I_x , I_y and I_t .

Step 4: By solving the 2-by-2 linear equations for each pixel and comparing the eigenvalues with the threshold to calculate the u and v .

- *Algorithm 4: Farneback (based on polynomial expansion)*

The Farneback algorithm [104] finds the displacement from low resolution to high resolution. It can compare with a pyramid structure, where optical flow estimation starts from one-point convergence and goes to many motion points convergences. The Farneback

algorithm finds displacement with the polynomial expansion method. The polynomial uses to approximate the neighborhood of each pixel. The expression of the local coordinate system is as follows:

$$f(x) = x^T Ax + b^T x + c \quad (4.7)$$

Where in equation 4.7 [104], “ A ” is a symmetric matrix, “ b ” is a vector, and “ c ” is a scalar quantity. The algorithm-related step to estimate displacement gives in [104].

4.1.2 Experimental analysis

- *About Dataset*

The dataset used in work is NTURGB+D. The dataset divides into three categories, i.e., medical condition-based actions, daily actions, and joint actions (where more than two persons involve). There are 60 classes in the dataset from which results show videos from each category. The dataset has 56880 videos recorded using the Microsoft Kinect v2 sensor. RGB videos record in the provided resolution of 1920x1080. The classification of the dataset under three types is as shown in Figure 4.1.

The experiment shows 15 random videos of NTURGB+D. The category of actions with the name of videos in the dataset is shown in table 4.1. The results show in two ways: on one video of class drinking optical flow visualization and second PSNR-based results in table 4.2. In Figure 4.2, the video-6 frames used to show that (a) is the first frame of video-1, (b) is the third frame of video-6, and (c) shows the difference between the two frames, which depicts which changes are there between frame 1 and frame 3. If there is no difference means the whole difference frame is black. Every optical flow algorithm gives a velocity vector in the X direction, a velocity vector in the Y direction, magnitude, and orientation. Optical flow is the movement of brightness patterns in X and Y directions. It gives information about the movement of the object in one frame to another, which uses for object tracking and estimation of actions in videos. The discontinuities in the objects help to segment objects in a frame. The magnitude depicts the velocities of pixels, and orientation indicates the direction of objects.

The Optical Flow cannot determine at one point. There is always involvement of neighborhood elements. At one point, the brightness varies, but the patch has no motion [105]. The magnitude of frame 1, shown in Figure 4.3 for Farneback, Horn Schunck, Lucas Kanade, and Lucas-Kanade Derivative of Gaussian. More edges identify using Farneback, and no boundaries were detected using the Lucas Kanade derivative of Gaussian. In another two methods, Lucas Kanade is better as compared to Horn Schunck. In Figure 4.4, the

orientation shows using all-optical flow ways. Whereas, the finding orientations using Lucas Kanade outperformed the other methods. In Figure 4.4 (a), there is no visualization of the subject, and the same is in Figures 4.4 (b) and (c) also.

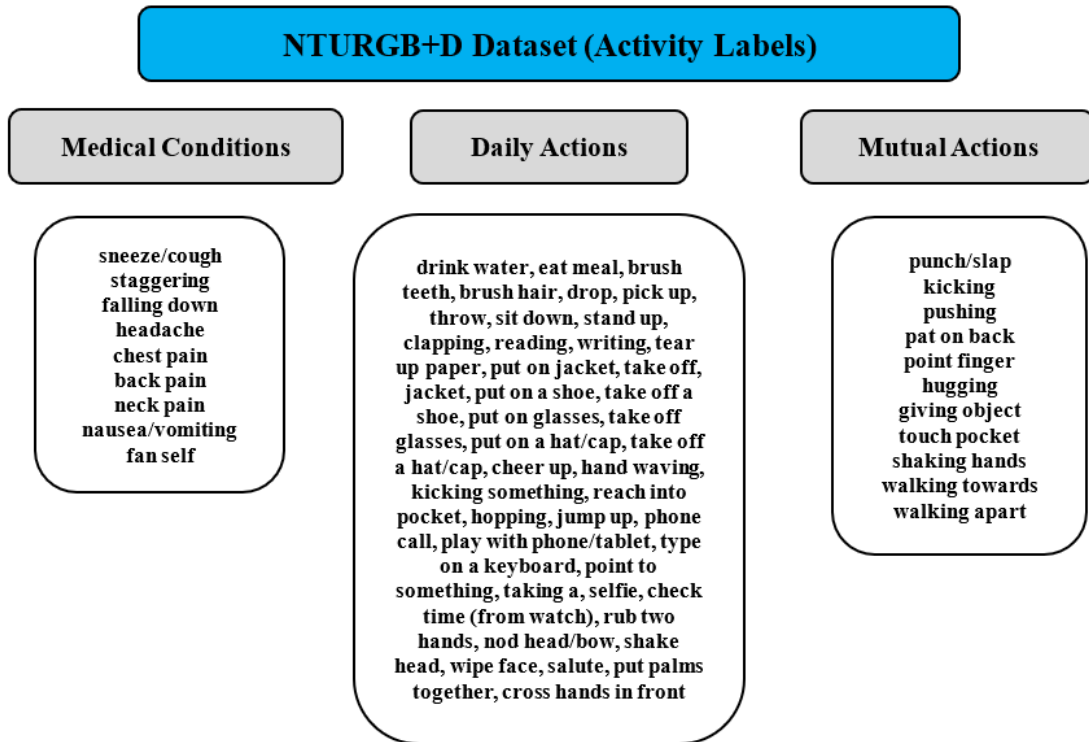


Figure 4.1 The activity labels under three categories are medical conditions, Daily actions, and Mutual actions of NTURGB+D

Figure 4.5, with frame 3 of video-6 magnitude optical flow, shows that Lucas Kanade is better than other methods. Figure 4.5 (a) detected extra edges, which are not required, and in Figures 4.5 (b) and (d), there is no visualization of subjects. The orientation provides information on the directions of subjects. The orientation of video-6 (Frame 3) shows in Figure 4.6, which depicts that there is no visualization in Figure 4.6 (a), (b), and (c) instead of Figure 4.6 (c). The magnitude and direction of optical flow is depicted in Figures 4.7 and 4.8 and 4.9, which demonstrate the distinction between the first and third frames of video 6. It shows a variation between frame 1 and frame 3 of video 6. And how the magnitude and orientation of optical flow affect frame-to-frame. From all four ways of optical flow, the Lucas Kanade is outperformed based on the frame and the difference of frames.

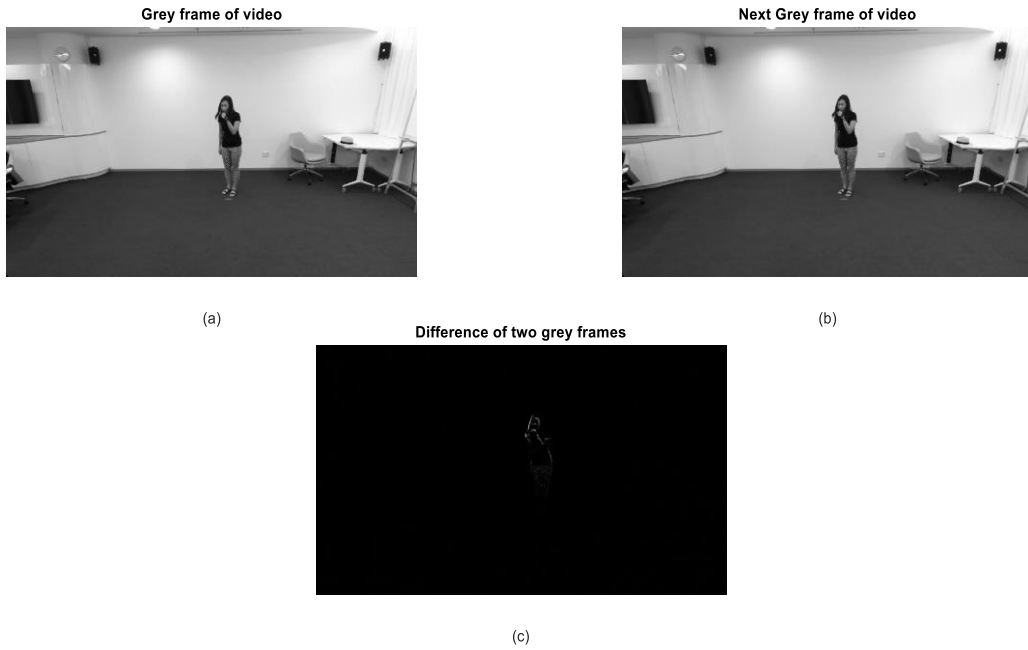


Figure 4.2 (a) Video-6 (First frame), (b) Video-6 (Third frame), (c) Difference between two frames

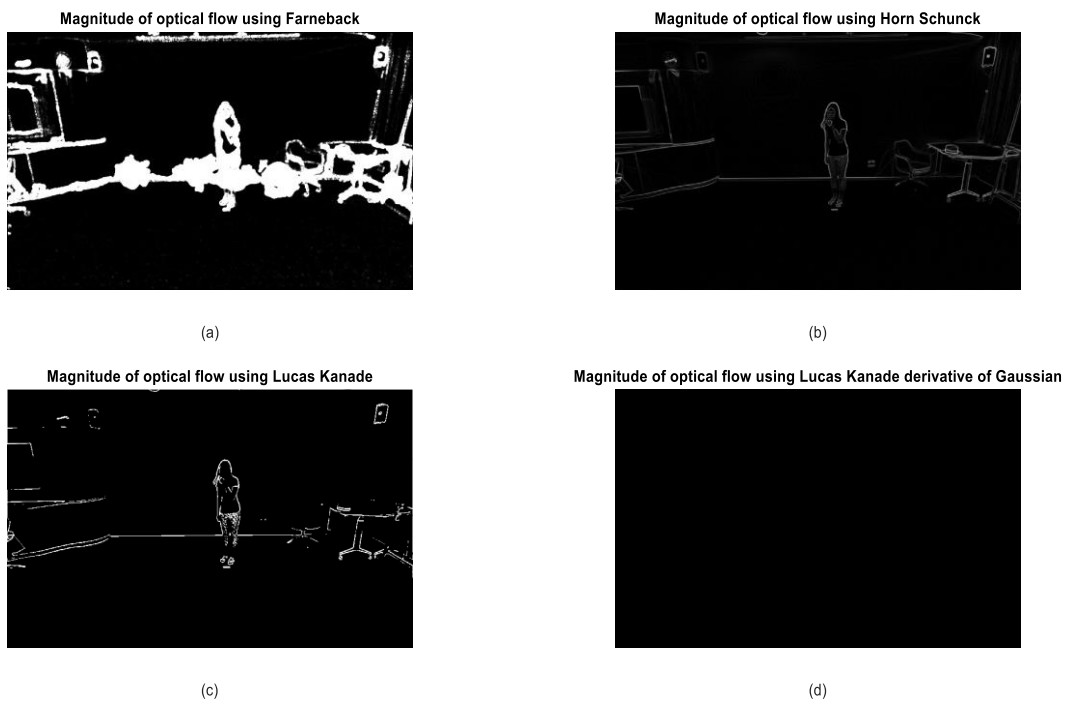


Figure 4.3 Video-6 (Frame-1) magnitude of optical flow a) Farneback, b) Horn Schunck, c) Lucas Kanade, and d) Lucas Kanade Derivative of Gaussian

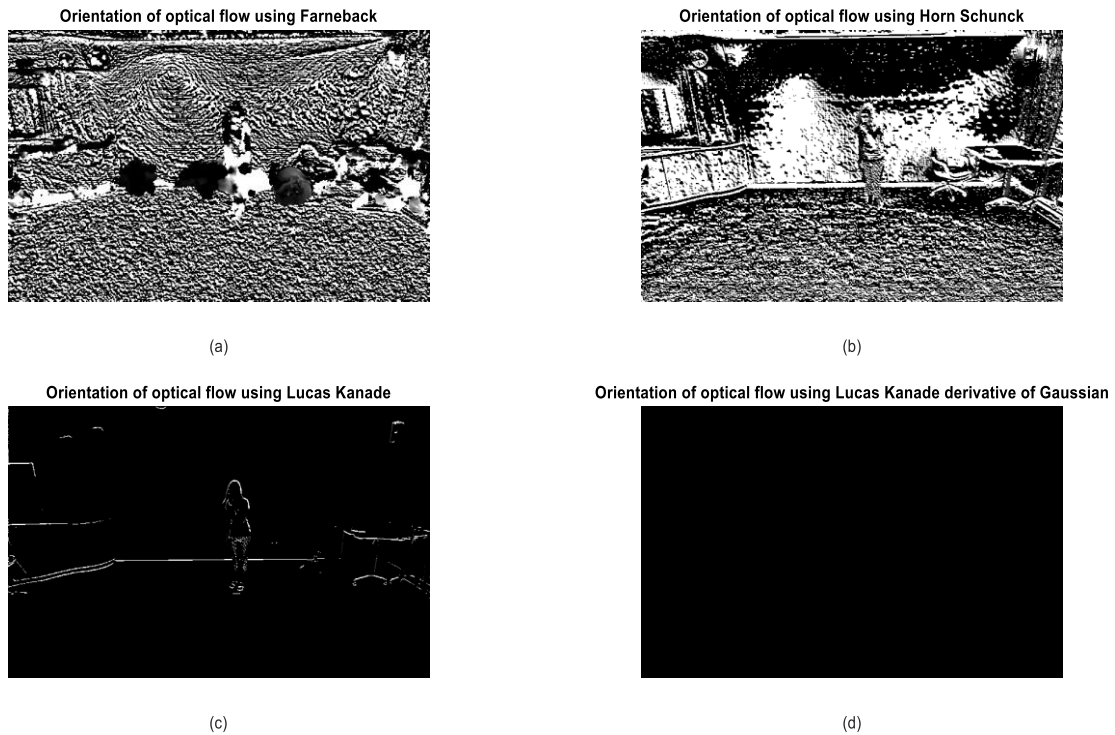


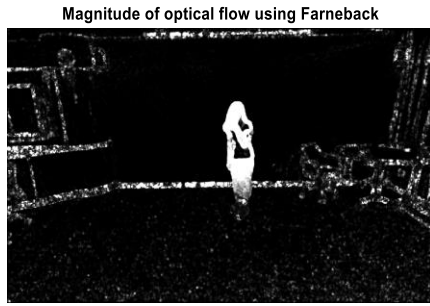
Figure 4.4 Video-6 (Frame-1) orientation of optical flow a) Farneback, b) Horn Schunck, c) Lucas Kanade, and d) Lucas Kanade Derivative of Gaussian

The experimental results show based on PSNR using equation 4.8. The PSNR is the peak signal-to-noise ratio. It is one of the evaluation metrics. Its gives analysis between signal and noise. It is the ratio between signal power and the noise power. The more positive the value of PSNR, the less noisy the data. The PSNR evaluates for 15 random videos of the dataset. The 5 videos used from each category as mentioned in table 4.1. The label of the video also gives in table 4.1. PSNR calculated on each video as mentioned in table 4.2 for all four optical flow methods, i.e., Farneback, Horn Schunck, Lucas Kanade, and Lucas-Kanade Derivative of Gaussian. With observation, the PSNR for OF using Farneback is negative. The more value of PSNR shows the better is the quality of an image. It shows that the variation in frames (Frame 3 and Frame 1 of videos) is the same as the variation in the magnitude optical flow ((Frame 3 and Frame 1 of videos) of all methods. Based on the observation, Lucas Kanade outperformed because the PSNR is the highest in many videos.

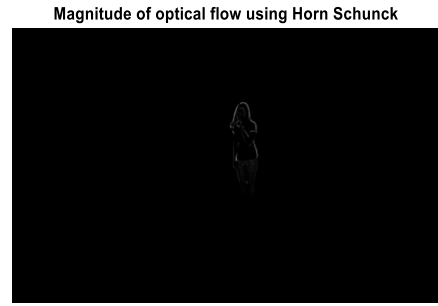
The PSNR calculates using the expression as

$$PSNR = 10 \log_{10} \left(\frac{peakval^2}{MSE} \right) \quad (4.8)$$

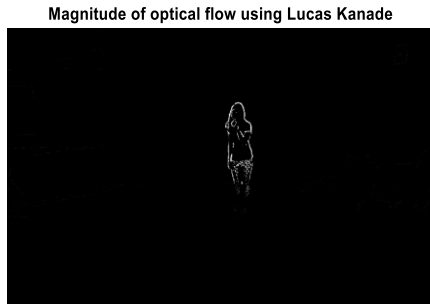
The MSE is the mean square error between the difference of magnitude/Orientation of frame 1 and 3 with difference between grey frame 1 and frame 3.



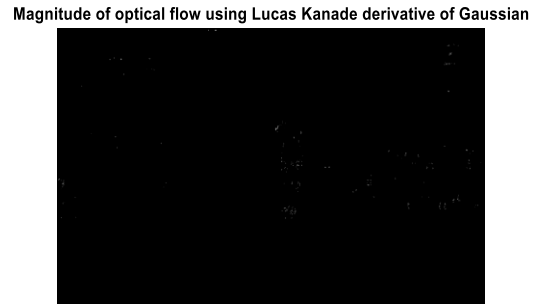
(a)



(b)

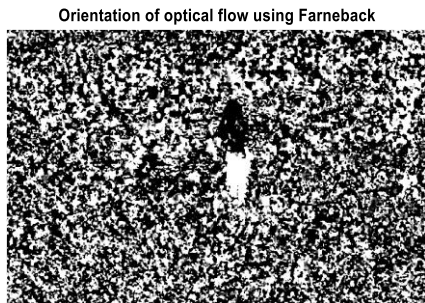


(c)

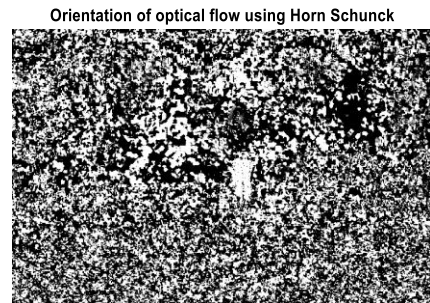


(d)

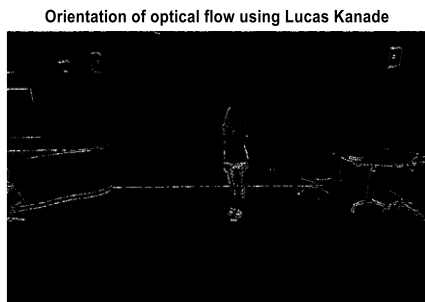
Figure 4.5 Video-6 (Frame-3) magnitude of optical flow a) Farneback, b) Horn Schunck, c) Lucas Kanade, and d) Lucas Kanade Derivative of Gaussian



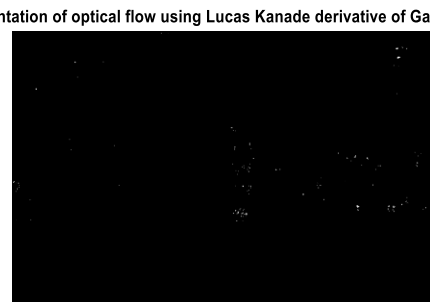
(a)



(b)



(c)



(d)

Figure 4.6 Video-6 (Frame-3) orientation of optical flow a) Farneback, b) Horn Schunck, c) Lucas Kanade, and d) Lucas Kanade Derivative of Gaussian

Table 4.1 Name of activities considered for analysis with video label and video label in NTURGB+D dataset

| Category of Actions | Name of activity | Video Label | Video label in NTURGB+D dataset |
|---------------------|------------------------|----------------|---------------------------------|
| Daily actions | throw | Video-1 | S001C001P001R001A007 |
| | drink water | Video-2 | S001C001P001R001A001 |
| | stand up | Video-3 | S001C001P001R001A009 |
| | reading | Video-4 | S001C001P001R001A011 |
| | put on glasses | Video-5 | S001C001P001R001A018 |
| Medical Conditions | nausea/vomiting | Video-6 | S001C001P001R001A048 |
| | sneeze/cough | Video-7 | S001C001P001R001A041 |
| | staggering | Video-8 | S001C001P001R001A042 |
| | chest pain | Video-9 | S001C001P001R001A045 |
| | back pain | Video-10 | S001C001P001R001A046 |
| Mutual Actions | pat on back | Video-11 | S001C001P001R001A053 |
| | kicking | Video-12 | S001C001P001R001A051 |
| | pushing | Video-13 | S001C001P001R001A052 |
| | hugging | Video-14 | S001C001P001R001A055 |
| | shaking hands | Video-15 | S001C001P001R001A058 |

Table 4.2 PSNR of each video using Farneback, Horn Schunck, Lucas Kanade, and Lucas Kanade Derivative of Gaussian

| Method | Farneback | Horn Schunck | Lucas Kanade | Lucas Kanade Derivative of Gaussian |
|----------------|-----------|----------------|-----------------|-------------------------------------|
| Video-1 | -33.1463 | -11.5332 | -11.5215 | -11.5492 |
| Video-2 | -30.8491 | -3.3579 | -3.4085 | -3.4121 |
| Video-3 | -30.5089 | -3.7779 | -3.7546 | -3.8372 |
| Video-4 | -31.3553 | 1.4849 | 1.196 | 1.3705 |
| Video-5 | -31.2982 | 1.0258 | 0.7899 | 0.9159 |
| Video-6 | -32.0707 | -13.2235 | -13.2216 | -13.2336 |
| Video-7 | -33.6824 | -10.3174 | -10.3111 | -10.3342 |
| Video-8 | -31.6548 | -12.7715 | -12.7623 | -12.7858 |
| Video-9 | -33.725 | -11.8957 | -11.8845 | -11.9109 |
| Video-10 | -35.2208 | -6.3572 | -6.3453 | -6.3951 |
| Video-11 | -31.8951 | -6.3589 | -6.3536 | -6.41 |
| Video-12 | -29.2746 | -16.3174 | -16.313 | -16.3265 |
| Video-13 | -29.7789 | -14.2096 | -14.1955 | -14.2236 |
| Video-14 | -30.1065 | -8.518 | -8.4931 | -8.5484 |
| Video-15 | -28.6731 | -9.5152 | -9.5015 | -9.5395 |

4.2 HOG (Histogram of oriented gradients) features

To detect the activity, the strongest point features are required. To achieve objective 1, the HOG (Histogram of gradients) has been calculated for some videos from set 1 of the datasets. These features are also kind of descriptors and focus on the object's shape in the image. The magnitude and orientation calculate for the detection of object present in the image. The HOG features calculate to understand the concept of object detection.

The HOG descriptor [108][108] basically gives information of silhouette contour. To calculate the HOG, the following steps involve:

- Step 1: The gamma correction of input data.
- Step 2: Calculate the gradient of an image.

Image gradients are calculated. The gradient is obtained by combining the image magnitude and angle. First, calculate G_x and G_y for each pixel value using the following formulas in equation 4.9.

$$G_x(r, c) = I(r, c + 1) - I(r, c - 1) \quad G_y(r, c) = I(r - 1, c) - I(r + 1, c) \quad (4.9)$$

where r, c refers to rows and columns respectively.

$$Magnitude(\mu) = \sqrt{G_x^2 + G_y^2} \quad Angle(\theta) = \left| \tan^{-1} \left(\frac{G_y}{G_x} \right) \right| \quad (4.10)$$

After calculating G_x and G_y , the size and angle of each pixel are calculated using the equation 4.10.

- Step 3: Calculate histogram of a gradient in cell.
- Step 4: Find the fundamental non-linearity of gradient descriptor i.e., spatial/ orientation binning.
- Step 5: Normalization of the block.

4.3 Summary

Motion estimation from videos for the understanding of motions can be one of the challenging tasks. In this work, optical flow visualization is analysed using four methods. Each technique helps calculate two main parameters: magnitude and orientation. The magnitude gives an understanding of brightness variation from pixel to pixel, and orientation gives the movement of directions. The optical flow visualization shows only one video, i.e., Video 6, mentioned in table 4.1 and table 4.2. It's also analyzed using PSNR that variation between two video frames is

the same in the optical flow magnitude of each method. Based on the visualization and PSNR, the motion estimating optical flow using Lucas Kanade outperforms the other three motion estimators. The Lucas Kanade can be the preference over others for applications like action recognition, 3D reconstructions, or video coding. The work can extend to using these features for motion estimations.

The histogram of oriented gradients (HOG) feature is utilized in the dataset for various videos. It is a sort of descriptor that is akin to the Canny Edge Detector and is used in computer vision and image processing to recognize objects. It evaluates the frequency of gradient directions in particular sections of an image. HOG descriptors focus on the structure or shape of an object. It outperforms any edge descriptor because it uses both gradient magnitude and angle to compute features. Generate a histogram using the gradient magnitude and direction for a region of the image. These features can use in activity recognition.

CHAPTER-5

HUMAN ACTIVITY RECOGNITION USING BILSTM NETWORK

Computer vision is an intricate discipline that is necessary for the automatic recognition, examination, and understanding of singular images or video sequences. Developing synthetic images from videos is one of the ambitious endeavors of actual computer vision in various fields such as industry, education, security, and consumer applications. In the process of identification or tracking, the two wide terms "action" and "activity" are often used in visual perception. There is still a distinction between these terms. An action is a basic movement pattern that a person completes in a short time, such as "lifting your arm", "bending", or "swimming". Alternatively, they are activities which the behaviour carries on for a long time. Two people can greet each other by shaking hands, a soccer team can successfully score a goal, and a group might take control of a plane and steal its contents [3].

To initiate an action or action recognition program, one may need to insert a video or photo sequence. Subsequently, extracting descriptions of functions and actions from the sequence is the following step. Finally, the interpretation of primitive acts is the last phase [3] represented in Figure 5.1. Figure 5.1 represents the various steps involved in designing an activity detection system. Research into human activity recognition continues to be a complex challenge due to the numerous variables involved. Variations in speed, movement, vision, and background noise can cause significant differences between classes. Furthermore, action perception is linked to situational elements such as scene features and interactions between humans and objects. Finally, the brief duration of action phases and the diversity of their shapes make it hard to model their progression [10].

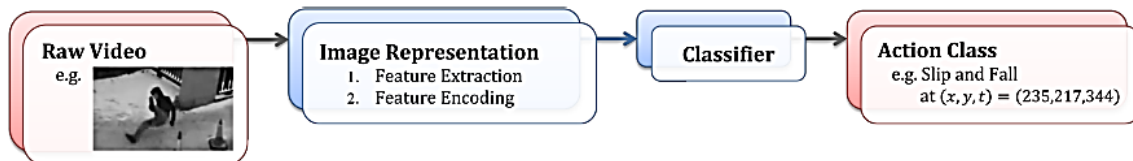


Figure 5.1 Process of Activity Recognition System [3]

5.1 Literature Survey

Recent developments in activity recognition have been of great interest, as evidenced by the wide range of applications that have been developed. Researchers have employed a number of creative humans and computer-based communication programs, such as managing presentation slides and providing instructions for employees to advance their knowledge and abilities. Additionally, memory-based attention systems have been utilized to enhance accuracy [16] can design large public spaces. The range of uses for this technology are vast, from the recognition of human activities like gaming, surveillance, and robotics, to more complex tasks [18]. One such application is an intelligent system designed for the elderly, which could be utilized to detect falls and notify family members in a timely manner [9]. This system would employ an orientation-based algorithm to track multiple individuals in nursing homes and identify individuals from a stored identity database [17]. Using deep learning, various frameworks from Hasan et al.[44] in which a system is developed to compute hierarchical features and a model to recognize behaviour based on computations. Researchers are investigating view-oriented activity recognition through the implementation of a hash table with multiple dimensions with a randomized voting system [34]. [21] recognition that is independent of viewpoint has proved to be an effective way of comprehending human activity through both motion and structural features [16]. This system provides a single learning mode that eliminates the need for camera measurements and incorporates a range of motion detection and crowd measurement subsystems. [17] use adaptive system to separate the background and foreground separation based on orientation feature vectors for recognition. An approach has been proposed that can handle video with moving backgrounds, lighting, and system runtimes, but these are also not well suited for high resolution [24]. The low-level feature vectors containing profile predictions across actions in response to actions and simple temporal objects shown in [31]. In [49], a multiview-based approach using deep networks was specified and design with LSTM by combining with CNN models. In [41] proposed a model of multiple feature expression and improved long short-term memory (LSTM) network performance. LSTM-based networks are the most accepted, and many HAR systems use RNNs. This is because this network operates on the basis of the past and the future. Similarly, in this work, the network was designed using his LSTM.

5.2 Feature Extraction for HAR

Feature extraction is the initial step in classification. A pre-trained network was utilized to change over each video into a sequence vector. Googlenet is a deep convolutional neural network having 22-layer, is used as a pre-trained system to extract the feature. The image input size for this specific design network is 224 x 224. Googlenet derives optimal results by representing video input frames as vectors while maintaining significant data of location. The transfer learning concept uses a worldwide average pooling layer to extract features form input video. The last four layers of the network are disregarded, because the network used for extraction of the features only, not for the classification. Each frame of the video represented into 1024 features and each video represented as the sequence matrix of [1024 X number of frames]. The graphical representation of the process to extract these features is illustrated in Figure 5.3.

5.3 BiLSTM Network for recognition using transfer learning

Deep learning is one of the emerging areas of research in the field of activity recognition. In this location, approaches based on deep learning have been developed, which have been broadly acknowledged as the most successful approaches for automatic action recognition. However, the traditional backpropagation model has been found to be associated with longer processing times due to insufficient backflow errors. To address this issue, the Recurrent Neural Network (RNN) has been proposed as an alternative model, which utilizes a gradient descent algorithm with comparatively lower computational complexity [109]. LSTM is one of the RNNs given by [109]. Mostly, LSTM networks are built using different cells and contain an input gate, an output gate, and a forget gate. In order to understand an LSTM network, one must consider the idea that cells store information over a period of time, and gates control the transfer of data between cells. If we were to envision an LSTM network, we would observe N blocks and M inputs [110]. There are various elements as shown below:

- *The block Input:* The current input x^t get combine with the LSTM output y^{t-1} using the block input in the last iteration. The equation 5.1 helps to do the corresponding calculation:

$$z^t = g(W_z x^t + R_z y^{t-1} + b_z) \quad (5.1)$$

Where in, W_z are the input weights and the R_z are the output weights, also b_z represent the bias weight vector.

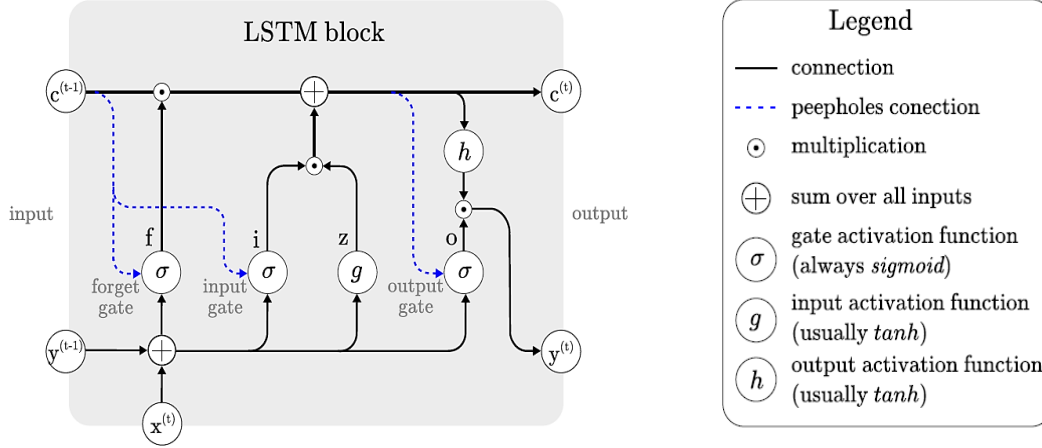


Figure 5.2 The LSTM block architecture [110]

- **Input gate:** It is used to merge the previous iteration's cell state with the previous iteration's inputs and outputs. Equation 5.2 displays the model employed to compute the input gate value.

$$i^t = \sigma (W_i x^t + R_i y^{t-1} + p_i \Theta c^{t-1} + b_i) \quad (5.2)$$

In the equation 5.2, the W_i, R_i and p_i are the weights which are used to update x^t, y^{t-1} and c^{t-1} respectively and b_i is called the bias vector by the input. In Equation 5.2, Θ is the vector pointwise multiplication process.

- **Forget gate:** It is use to determines the cell state information. The activation function for this gate is based on the input and output of the memory cell state in the previous step. For this calculation, Equation 5.3 below becomes:

$$f^t = \sigma (W_f x^t + R_f y^{t-1} + p_f \Theta c^{t-1} + b_f) \quad (5.3)$$

In equation 5.3, the W_f, R_f and p_f are weights that use to update the x^t, y^{t-1} and c^{t-1} respectively and b_f is called the bias vector corresponding to the input. In equation 5.2, Θ is the pointwise multiplication process of the vector.

- **Cell:** Formula 5.4 shows how to calculate the cell value.

$$c^t = (z^t \Theta i^t + f^t \Theta c^{t-1}) \quad (5.4)$$

- **Output gate:** After combining the input, output and cell values, we calculate the output of the gate using the given formula 5.5.

$$o^t = \sigma (W_o x^t + R_o y^{t-1} + p_o \Theta c^t + b_o) \quad (5.5)$$

- **Block output:** Finally, use the cell values and output gate values to calculate the block

outputs. The corresponding equation 5.6 is used.

$$y^t = g(c^t) \Theta o^t \quad (5.6)$$

Where σ, h and g used to pointwise non-linear activation functions. The sigma is the logistic sigmoid function, and g and h are $\tanh(x)$ functions, respectively [110]. Figure 5.2, Two LSTMs in opposite directions are combined into a network to form a BiLSTM network.

In BiLSTM, the output of a one LSTM layer is determined by the trace from the previous vector, the present vector, and the direction of the future vector. This allows the network to retain more information and make use of future data. Furthermore, this type of network is capable of transitioning between continuous data and video signals [111].

This work utilized a BiLSTM network, resulting in an excellent activity accuracy level. It can be said that the BiLSTM network employed in our model captured the long-term bidirectional relationships embedded in all parts of the video by iteratively passing through the vector sequences [111].

The extracted features become the input for the network sequence layer which is then used for action recognition (see Figure 5.3). The architecture of the layered LSTM network is illustrated in Figure 5.3. It consists of several layers: a sequence input layer, a BiLSTM layer, a dropout layer, a fully-connected layer, and a softmax layer. The last classification layer produces the class labels as the output. The sequence input layer has an input size that matches the size of the feature vector. Following this, a BiLSTM layer with 200 hidden units is added, followed by a dropout layer. The 'Output Mode' of the Bi-LSTM layer is set to 'Final', which implies that only one label is used for each sequence (as seen in Figure 5.3). The BiLSTM layers are employed to explore the long-term dependencies of the time series features of each video. The dropout layer is essential to the network training, as it helps improve the model performance by reducing the overfitting. The softmax layer is used to run neural networks for multiclass functions and helps determine the multiple classes in an image or video.

5.4 Experimental Analysis

5.4.1 Implementation details

A pre-trained configuration of GoogLeNet is used to convert video frames into feature vectors. A large database of human movements with 2 GB of video data and 7000 clips across 51 classes was used for this purpose. A Convolutional Neural Network (CNN) was used to compute

the feature output for each video frame. The video was divided into sequences as feature vectors, which were extracted from the GoogLeNet global mean layer. After features were extracted, a network was designed to recognize activities. This network was trained and tested on all videos, and the video was partitioned into 90% for training and 10% for validating the HAR network.

The Bi-LSTM units used in the network were 200 with a stack size of 16. The entire network was trained from start to finish using an initial learning rate of 0.0001 and the Adam optimizer. A dropout layer of 0.5 was utilized to prevent overfitting. The implementation was executed on an i5 processor, NVIDIA GEFORCE, 8 GB RAM, and 1 TB SSD.

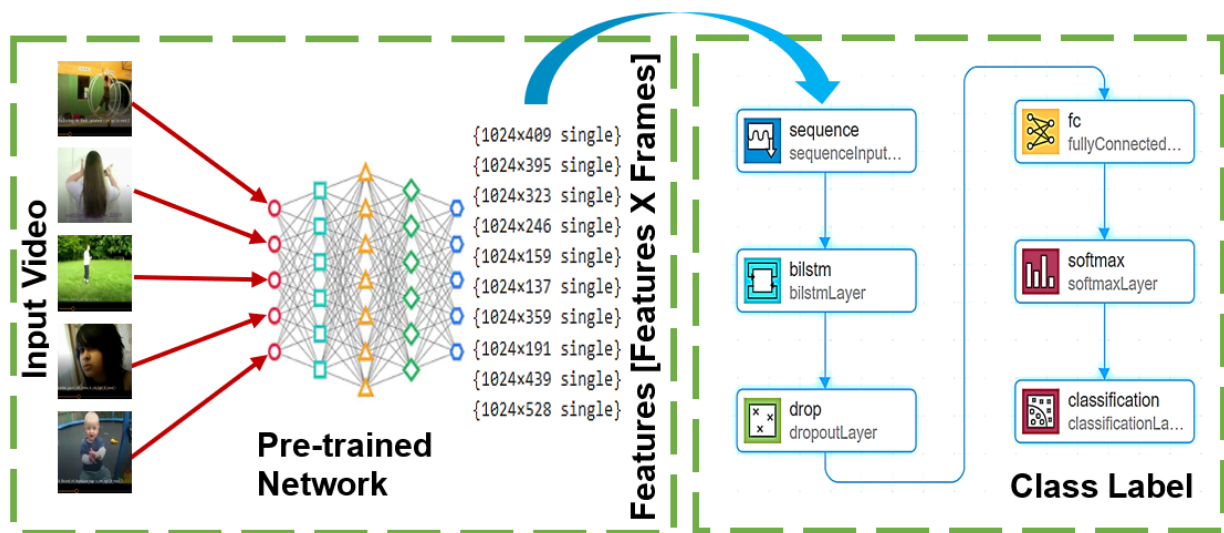


Figure 5.3 Network model for HAR

5.4.2 About dataset HMDB51

The Human Movement Database (HMDB51) is a comprehensive collection of videos covering a wide range of human movements. It includes 6849 clips organized into 51 action categories, each containing a minimum of 101 clips. These 51 categories can be further divided into five main groups:

- The class name, smile, laugh, chew, speak includes under category of ordinary facial expressions.
- Smoke, eat, drink includes under facial actions by deception of something
- Cartwheel, clapping, climbing, climbing stairs, diving, falling down, backhand flip, handstand, jump, lift up, run, run, sit down, sit down, distract, get up, turn, go, wave are

related to typical body movements

- Hair brushing, gripping, drawing a sword, pulling a golf ball, hitting something, kicking a ball, picking, pouring, pushing something, cycling, horseback riding, shooting a ball, shooting a bow, shooting a gun, swinging a baseball bat, exercise with the sword, throwing is related to physical movement with object contact
- Fencing, hugging, kicking, kissing, punching, shaking hands, fighting with a sword relates to physical movement of human contact

Figure 5.4 shows a detailed distribution of databases, Figure 5.5 illustrates the breakdown of databases according to a range of parameters, which include physical appearance, camera action, camera viewpoint, and clip quality. This database was then used to train and validate a system that is able to predict classes.

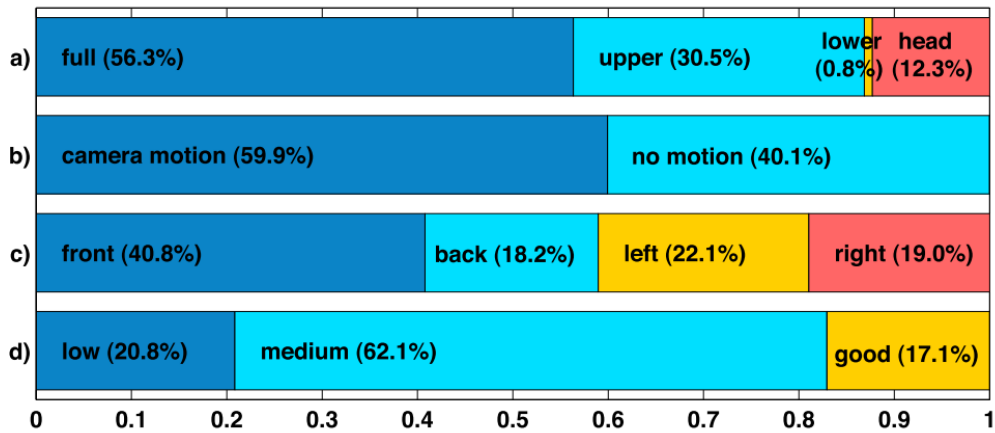


Figure 5.4 The HMDB51 contains different categories based on a) body parts that are visible, b) the motion of the camera, c) the point of view of the camera and d) the quality of the video clip.

[112]

5.4.3 Conduct an analysis for 10 classes of HMDB51

Results were obtained in 10 classes: brushing, stair climbing, cartwheeling, catching, chewing, clapping, climbing, diving, drawing the dribbling, and sword. 1150 videos were used in the training and testing process. The progress of the training is depicted in Figure 5.5. In the starting, the accuracy improves rapidly up to 100 iterations. After 100 iterations, it increases very slowly and reaches a maximum value of 93.04%. The second graph in Figure 5.6 is the change in loss with iteration. At the 0th iteration, the validation loss is very high, but from the 0th to the 200th iteration, there is a marked decrease in the loss. It then decreases very slowly, ending at

0.3245.

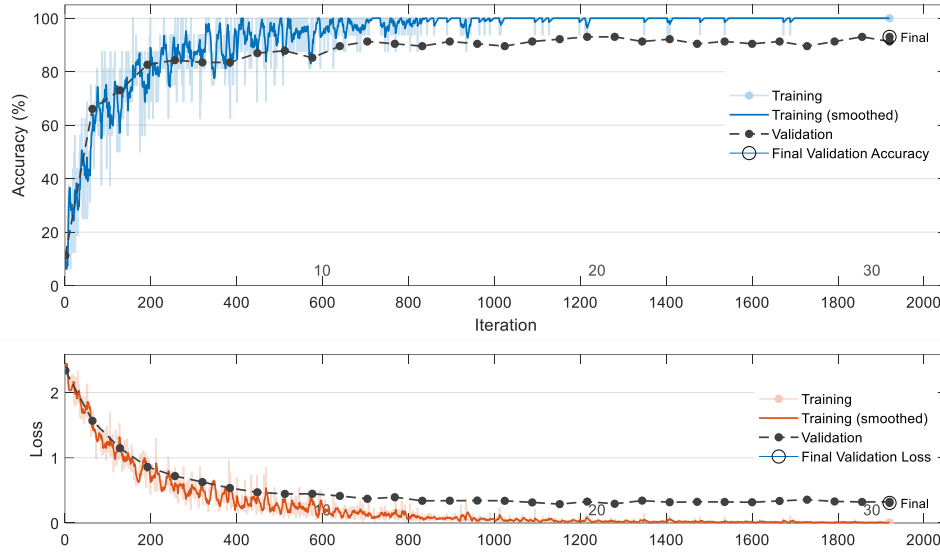


Figure 5.5 a) The progress of training using HMDB (10 Classes) is represented by a blue-colored continuous line, b) The changes in loss are displayed by a red-colored continuous line

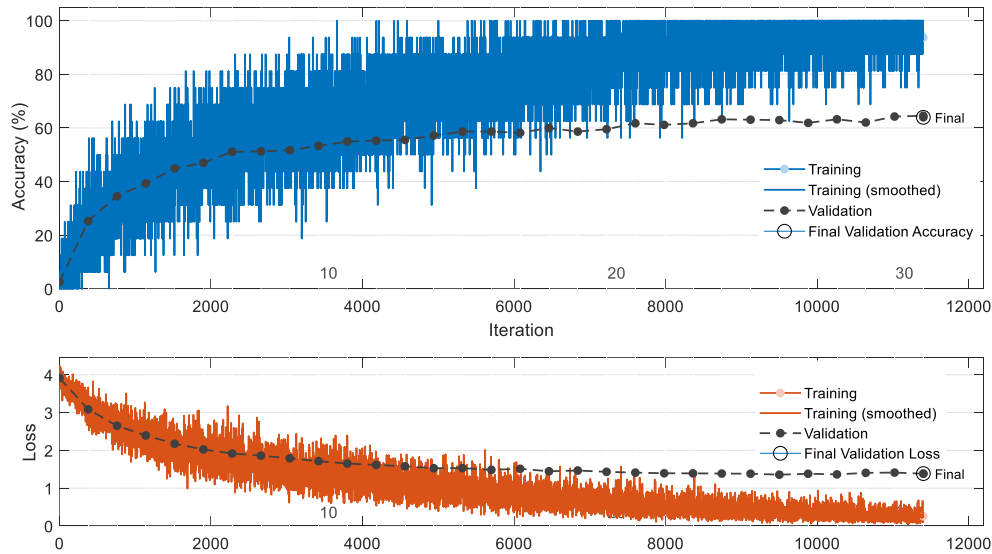


Figure 5.6 a) Utilizing HMDB (51 Classes) for tracking the progression [Blue colored solid line], b) Alterations in the loss rate [Red colored solid line]

5.4.4 Conduct an analysis for 51 classes of HMDB51

In Figure 5.6, the experiment was performed using the full data set, populated with 51 classes. Observation charts are shown in Figure 5.6 as well as Figure 5.5. In the graphs, the first related to accuracy and the second related to iteration-related validation loss. The precision of the

results is not comparable to that of the previous experiment, given the large number of classes. Up to the 200th iteration, the accuracy exceeds 40%, after which it increases slightly and ends at 63.96%. Same way, the loss-based validation observation tapers off after the 2000th iteration, ending at about 2000. 1.56. Considering all classes in the HMDB51 dataset, the accuracy is 63.96%. When compares proposed method and compared with other, it shows outperformed shown in table 5.1.

5.4.5 Comparison with other state of art methods

The proposed work is compared with other existing state-of-the-art methods in Table 5.1.

Table 5.1 Comparison with other method

| Reference | Accuracy (%) |
|--|--------------|
| Zhu et al. 2018 [113] | 53.8 |
| Tran et al. 2017 [114] | 54.9 |
| Simonyan and Zisserman 2014 [115] | 55.4 |
| Srivastava, Mansimov, and Salakhutdinov 2015 [116] | 44.1 |
| Sun et al. 2015 [117] | 59.1 |
| Wang et al. 2013 [118] | 57.2 |
| Bilen et al. 2018 [119] | 57.2 |
| Pan et al. 2021 [120] | 49.2 |
| Our Method | 63.96 |

5.5 Summary

In our work, we presented a model based on BiLSTM and compared its performance to the other methods discussed in Table 5.1. All the systems require the same optimization, with the model achieving a high recognition rate when considering only 10 classes. However, upon increasing the number of classes to 51, the accuracy decreased to 63.96%. It is crucial to understand that accuracy relies heavily on factors such as the feature vector, the number of layers, and tuning the model's hyperparameters. Deep learning concepts are the foundation for all these components.

The system can be designed in the near future using multiple training videos after the hyperparameters have been identified. This design is based on supervised learning. In addition, models can be examined to determine if unsupervised learning is an option. The data set used in this study consists of a single-view dataset. By expanding this model, the best model can be attained. This is also the case for datasets with various perspectives.

CHAPTER-6

ACTION RECOGNITION FOR MULTIVIEW USING NTURGB+D DATASET

Human activity recognition is a recent area of research for researchers. Activity recognition has many applications in smart homes to observe and track toddlers or oldsters for their safety, monitor indoor and outdoor activities, develop Tele immersion systems, or detect abnormal activity recognition. Three dimensions (3D) skeleton data is robust and somehow view-invariant. Due to this, it is one of the popular choices for human action recognition. This chapter includes proposed using a transversal tree from 3D skeleton data to represent videos in a sequence. Further proposed two neural networks: convolutional neural network recurrent neural network_1 (CNN_RNN_1), used to find the optimal features and convolutional neural network recurrent neural network network_2 (CNN_RNN_2), used to classify actions. The deep neural network-based model proposed CNN_RNN_1 and CNN_RNN_2 that uses a convolutional neural network (CNN), Long short-term memory (LSTM) and Bidirectional Long short-term memory (BiLSTM) layered. The system efficiently achieves the desired accuracy over state-of-the-art models, i.e., 88.89%. The performance of our proposed model was compared with existing state-of-the-art models using the NTURGB+D dataset, which is one of the largest benchmark datasets for human activity recognition. The comparison results indicate that our proposed model has achieved better performance than the state-of-the-art models.

6.1 Literature Survey

Action is when we do something, especially when dealing with anything like an object or human. The primary purpose of any human activity recognition system is to identify activities that are taking place in a video in real-time. This recognition of human activities can help in the surveillance of public venues like airports and train stations, and it can also be employed to monitor patients, kids, and seniors [121]. Vision-based activity recognition systems highly impact various motivating application domains, like behavioral biometrics, Content-based video analysis, security and surveillance, interactive application and environment, animation and synthesis [3]. In behavioral biometrics, various approaches are based on Fingerprint, Face, or Iris and are used to recognize human-based physical or behavioral cues. In this approach, the subject's cooperation is

not required and only needed to know the subject's activity. Gait recognition [4] could be the most challenging application area. After all, a person walking characteristics can identify the person through closed-circuit television (CCTV) footage because everyone has a distinct walking style like other biometrics. Today, with fast-growing technology, people can share and search multimedia content, such as images, music and Video. Searching for desired content is very challenging for a retrieval system to find a subset of objects with similar content [5]. Summarizing and retrieving consumer content, such as general activities like sports or cooking videos, are one of the most commercial applications under content-based video analysis. The team was creating a visual surveillance system that could see the movement of items in a certain area and learn from the patterns of activity. This system would include motion tracking, activity identification, and incident recognition.

An area can be significant to observe from a single camera, so many such sensor units use around the site. Cameras are attached to poles, trees and buildings for an outdoor setting. The indoor setting involves attaching to walls and furniture [6]. Research into intelligent surveillance has increased due to its successful deployment in a number of areas, such as public spaces, airports, railway stations, shopping centers, and military installations. Additionally, intelligent healthcare facilities have been utilized in assisted living facilities to detect falls in elderly people [122]. Often, the objective is to identify, recognize, or become aware of interesting occurrences, classified as dubious events, anomalous behavior, uncommon activities/events/behavior, or strange behavior [7].

For such activity, using CCTV cameras to record or observe scenes the user has become ubiquitous. Although recording videos through cameras is cheap, affordable and popular in today's scenario. However, the agents for observing outliers and analyzing the footage are also limited and unreasonable. Wherever video cameras use in the room, they experience poor monitoring due to genuine reasons like the fatigue of the observer. Due to long monitoring hours, the operator can skip noticing suspicious activity, which is generally of short duration. This application comes under security and surveillance because detecting unusual activity at the right time is essential. One of the difficulties of designing a human-computer interface lies in the interaction between humans and computers in interactive applications and environments. In today's scenario, smart devices are capturing data and analyzing users. The relevant information can extract from the activity tracker for activity recognition. The framework explores fog computing to the cloud for

reducing computation proposed in [123]. An interactive domain such as smart rooms responding to a person's gesture can directly or indirectly benefit the user [124]. Such as music according to the user's mood when entering the room. Animation and synthesis require an extensive collection of motions the animator uses to make high-quality animation or movies. Any application can relate human motion to any environment, including training military soldiers, firefighters and other personnel [3] [125].

Human activity recognition consists of preprocessing, segmentation, Feature extraction, dimension reduction, and classification. However, various data modalities are available to detect action from activities. Instead of other modalities, 3D information uses to track movement. 3D information contains coordinates value that helps to track body joints efficiently [49]. Nowadays, recording videos at various angles is called multiview data. Multiview learning [126] [127] is essential for action recognition. The camera can employ at any angle for recording actions. There is a requirement for the system to detect activities, which can handle many views for identifying actions. Nowadays, deep learning-based methods have accomplished importance in human activity recognition. The recurrent neural networks-based system is considered adequate for sequential data handling [41] and specially designed with LSTM [109], or BiLSTM [128] layered recurrent networks.

Gaining awareness of the most recent progress in activity recognition is a captivating endeavor. The skeleton data has joints describing the body's movement and pose. In multiview action recognition, 3D information can prefer and get more attention. In [129], the hand-crafted feature has been calculated and given to the CNN-based model for skeleton-based action recognition. The work has high computational complexity based on the state-of-the-art comparison. In other words, a convolutional network uses additional features like joint distribution trajectories [130]. Instead of CNNs, recurrent neural networks (RNNs) can prefer because they store the information based on time dependencies which is the essential information in action recognition. In [131] shows multi-model strategies using LSTM. LSTM has proven to be a good choice for data where time-based information is a concern. A few selective frames can choose to find features in the proposed method. Finding features from a smaller number of frames can reduce the computational complexity of the system because frames in videos have replicated activities also. The method [132] used a deep neural network and suggested selecting a keyframe. The LSTM is often applied [133] differential LSTM, which proposes feature enhancement. A densely

BiLSTM network is presented in [134], which outperforms spatial and temporal data. In contrast to all these methods, the proposed method in this work with two neural networks designed with LSTM and BiLSTM layers.

This section includes a survey of the current knowledge, including essential findings based on applications. Providing a machine that can detect or recognize activities from videos through which a human can understand or act based on activity. Main contribution towards action recognition with the proposed work:

- Technique for preprocessing of 3D skeleton information.
- The deep neural network can use as a pretrained network for multiview data.
- Network which can use for feature reduction.
- Deep neural network for classification of action from 3D skeleton information.
- This proposed method is independent of multiview and multiple subjects.

6.2 Proposed research methodology

The NTURGB+D dataset has various data modalities from these modalities, 3D joint information used in the proposed methodology.

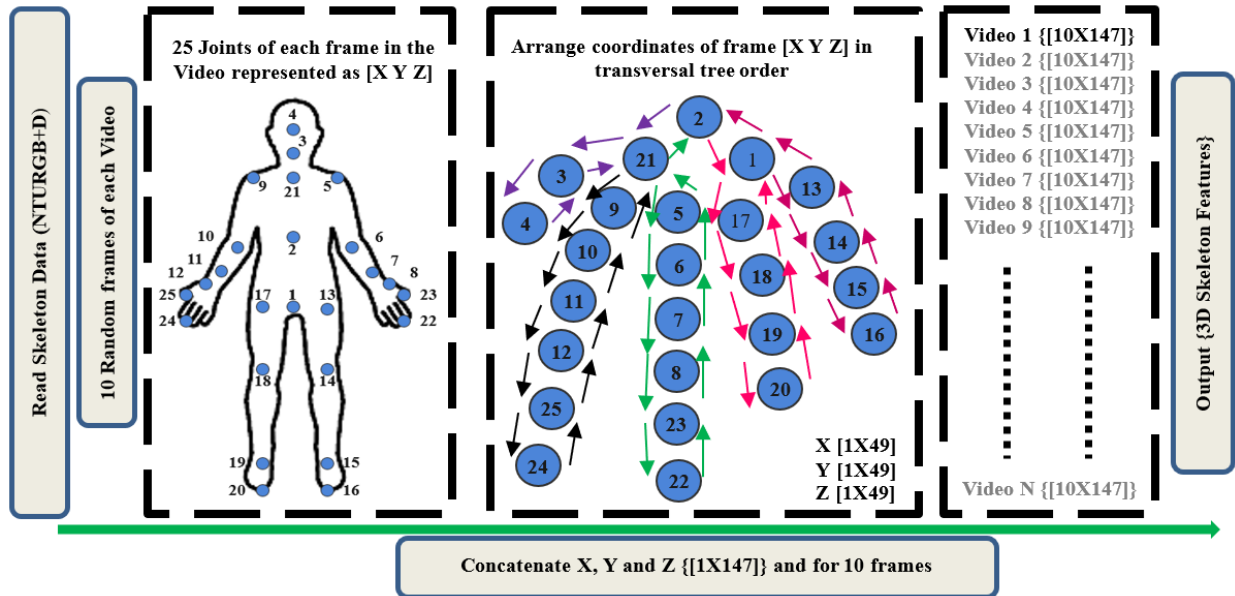


Figure 6.1 3D Skeleton features using the transversal tree

The information in a skeleton file from which three coordinates, X, Y and Z, have been extracted and used for further representation in the form of optimal features. The steps for extracted

3D data has given in Figure 6.1.

The skeleton file has various information, i.e., body ID, Clipped edges, hand left confidence, hand left state, hand right confidence, hand right state, lean X, lean Y, joint count and Joints. Take further on Joints: X, Y, Z, depth X, depth Y, color X, color Y, orientations and tracking details are available. Each frame of the Video has information on X, Y and Z, shown in Figure 6.2.

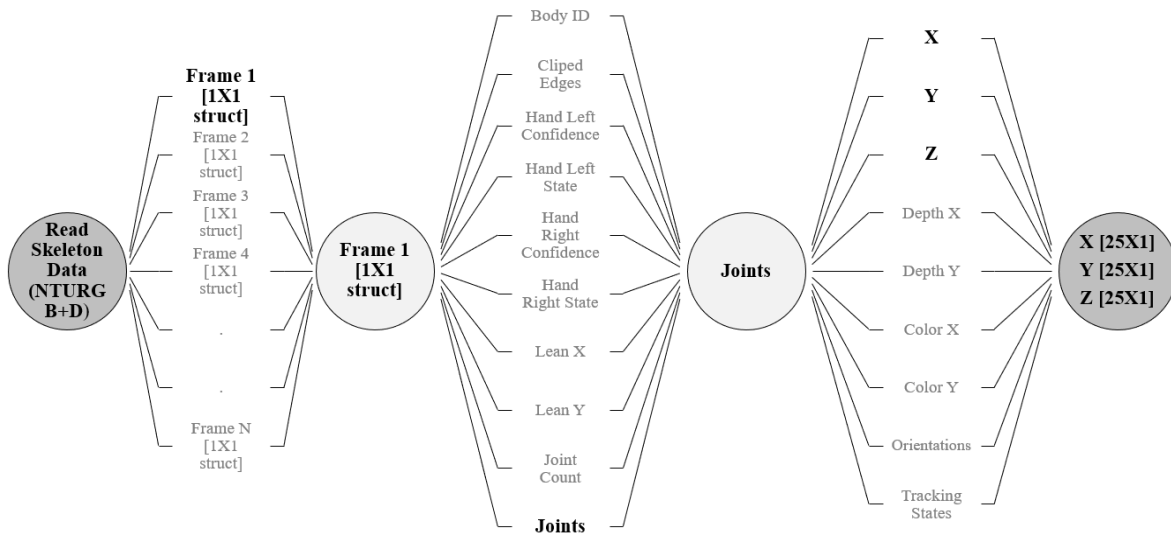


Figure 6.2 Step for 3D coordinates of Skeleton 3D NTURGB+D dataset.

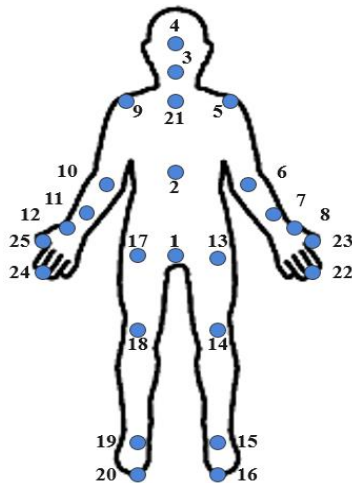


Figure 6.3 Joint location of the human body of Skeleton 3D data [25 Joints]

The X, Y and Z are the 3D coordinates values of joints. There are 25 Joints body positions for activity recognition, as given in Figure 6.3. The dimension of each vector is [25X1]. Each vector is rearranged in the tree so that network can train in a more appropriate form.

The technique being discussed includes three major elements, as can be seen below.

6.2.1 3D Skeleton Pre-processing (Representation of 3D Skeleton Data into Sequences Using Transversal Tree)

For human activity recognition, the features play an essential role. The skeleton sequences of NTURGB+D contain Skeleton 3D data for all videos of 60 classes. Each Video from the dataset has a different number of frames. The random ten frames have been considered for the same size output to make the Skeleton feature. For example, if any video contains 100 frames. Then, the index of frame selection is [1 11 21 31 41 51 61 71 81 91]. Each frame contains joint information containing X, Y and Z coordinates values. The X, Y and Z are the 3D coordinates of skeleton sequences. The 25 joints correspond to which point of the human body, as given in Figure 6.3. The size of each X, Y and Z is [25 1]. All coordinates are arranged as transversal tree orders [129], in which the index of the joint number has stores accordingly. The order of making a new sequence is [2 21 3 4 3 21 9 10 11 12 25 24 25 12 11 10 9 21 5 6 7 8 23 22 23 8 7 6 5 21 2 1 17 18 19 20 19 18 17 1 13 14 15 16 15 14 13 1 2]. The total number of joints available is 25. To give the network a sequence with that it can train efficiently. Make that number of the joint index 49 for each X, Y and Z as per the transversal tree. Concatenate the ten frames X, Y and Z, 3D skeleton data and size becomes [10X49] where 10 is the number of frames chosen from each Video and 49 is the X, Y and Z data index. There is a requirement to give a pattern for system learning. Joints arrange to give relative motion information of coordinates. The joints arrange in the manner of body parts, i.e., the torso, right arm, left arm, right leg and left leg. Each Video from the dataset has represented a new skeleton 3D of fixed size [10X147]. These are the feature which is the input to CNN_RNN_1.

6.2.2 CNN_RNN_1 for Optimal Features

In this work, multiple LSTMs and CNNs uses for human activity recognition. Generally, researchers use a pre-trained network to find optimal features. There is no such pre-trained network that acts optimally for the NTURGB+D dataset. The LSTMs-based model is contextually dependent on the temporal domain. Moreover, CNN-based models focus on spatial information. Temporal and spatial information are essential features for action classification using Videos. CNN_RNN_1 train to find optimal features for another network called CNN_RNN_2. First, the

CNN_RNN_1 train using a dataset then the network is used to find optimal features. Input videos dataset fix for CNN_RNN_1 as well as CNN_RNN_2 for training purposes. The video dataset for validation and testing is distinct from training for CNN_RNN_1.

The CNN_RNN_1 consists of 20 layers, including LSTMs and CNNs with other essential layers. After giving the input, the first layer is the folding layer, which converts a batch of an image sequence to a collection of images and converts the sequence for the next layer to convolution. The convolutional layers that follow one after another have 32, 48, and 64 kernels in order. After each Convolutional layer, there is a ReLU layer and a Maxpooling layer. The ReLU layer uses to operate any value zero if the value is less than zero.

Furthermore, the Max pooling layer uses to down sample the input and half the number of samples is the output. The unfolding stage of the process reverses the sequence folding of the input data, returning it to its original sequence. After the unfolding layer, a flattened layer converts data into a single column. The flattened sequence passes to LSTM layers. Each LSTM layer ended with a dropout layer, half the sample length. The number of neurons is the same in the LSTM layer, which is 128. The dropout layer with a probability of 0.5 to avoid network overfitting during learning. The last three layers fully connect with the number of neurons, same as the number of classes, i.e., 60, Softmax layer for determining the probability corresponding to each class and classification layer for assigning class as per probability based determined with softmax layer.

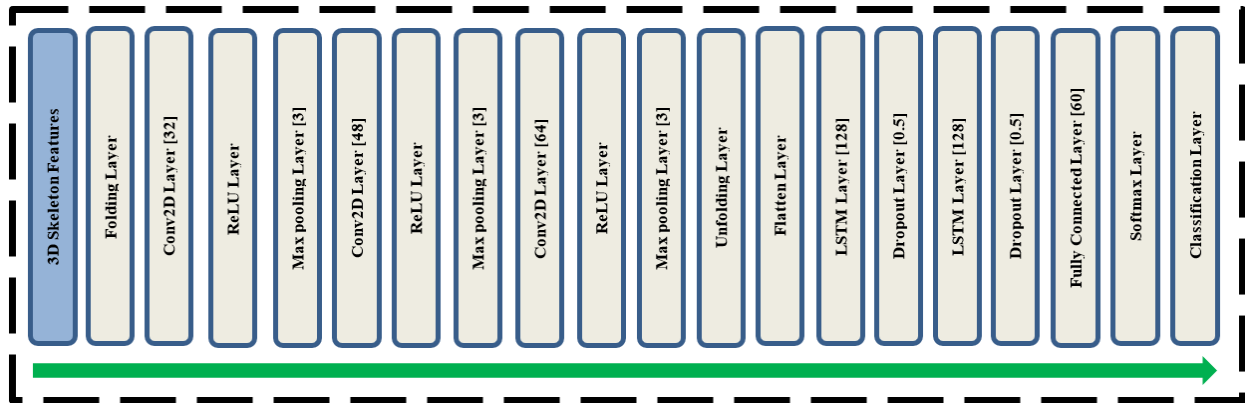


Figure 6.4 Architecture of CNN_RNN_1 for feature reduction

6.2.3 CNN_RNN_2 for Action Classification

CNN_RNN_2 is proposed for the classification and is first required to load the trained proposed CNN_RNN_1. With CNN_RNN_1, optimal features calculate for the same training,

validation and testing data used for CNN_RNN_2. The dimension for Skeleton 3D features of each Video is [128 1], as mentioned in Figure 6.4. The sequence input gives to the flattened layer. In CNN_RNN_2, the BiLSTM layer uses as in Figure 6.5. The two LSTM networks are connected in opposite directions to make BiLSTM. Our model utilizes the BiLSTM network to capture long-term bidirectional relationships in the embedded vector sequence across the video by traversing back and forth multiple times [111]. BiLSTM layer uses to store time dependencies of data and preference for classification of CNN_RNN_2 used here to classify the action from videos.

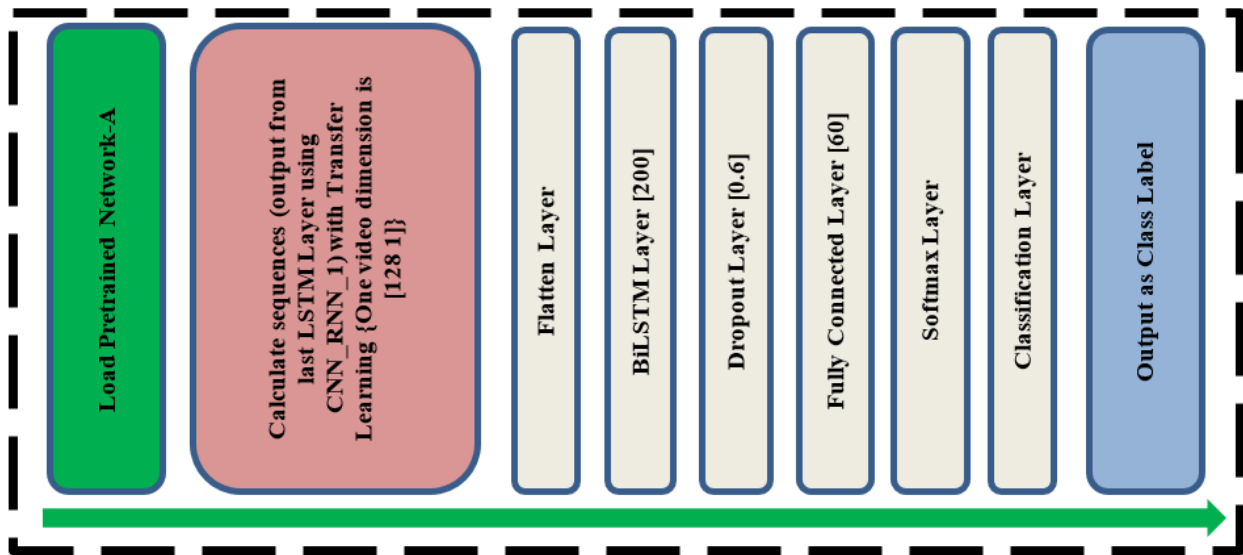


Figure 6.5 Architecture of CNN_RNN_2 for classification

6.3 Experiment Analysis

6.3.1 About Dataset NTURGB+D

The evaluation of the NTURGB+D dataset was addressed by a proposed method. The ROSE Lab was established in a collaborative effort between Nanyang Technological University in Singapore and Peking University in China. Four data modalities are available i.e. Depth map, 3D connection information, RGB frames, IR sequences. In this work, the 3D joints information only uses. The data set includes the 3-dimensional coordinates of 25 major body joints, as depicted in Figure 6.2. The major body joints help to detect and track the movement of each body part of the human body. The dataset contains 60 actions classes-based information and 56880 videos recorded using the Microsoft Kinect v2 sensor. There are 40 distinct subjects between the age of 10 to 35 years. The dataset videos recording at three different angles, i.e., 45° , -45° and 0° , as

shown in Figure 6.6. The dataset is divided into three main categories: 40 activities regarding day-to-day activities, 9 pertaining to human health, and 11 that are mutual activities. This dataset has a variation in the number of subjects and ages of subjects [53].



Figure 6.6 Three views in NTURGB+D (side view (+45°), front view (0°) and side view (-45°))

6.3.2 Implementation Details

The experiment performs on the system with a 1.6 GHz Intel Core i5-4200U, 8GB RAM and 1TB SSD running a Windows 10 with the 64-bit operating system. There are CNN_RNN_1 and CNN_RNN_2; both trains, validate and test with the dataset. The CNN_RNN_1 has 20 layers which train by using a dataset. No pre-trained network is available for this application to find optimal classification features. The filter size is [3 3] for the convolutional 2D filter. The Adam optimizer uses during training with a learning rate of 10^{-4} . The minibatch size is 128, with the number of iterations per epoch being 32 upon max epoch 500. The same hyperparameter has been considered for CNN_RNN_2 also. The experiment performs on a total of 960 videos of NTURGB+D. It includes three angled-view videos for all 60 classes.

6.3.3 Results

The arrangement of joints aims to represent the 3-dimensional coordinates in some pattern. So, networks can learn the pattern in a way; it can identify the activities efficiently. The captured 3D information preprocess for the network. After preprocessing, the 20-layer CNN_RNN_1 network train with some videos. The same CNN_RNN_1 uses to calculate optimal features of which dimension is much less than the input data. It can use as a feature reduction technique to represent a video in a single column of length [128 1]. When all the videos are the same size as [128 1], these features are the input to the second network called CNN_RNN_2. CNN_RNN_2 is

used to classify the action from 3D information. The conclusion is that there are two networks, one for feature reduction technique and another for classification. The experimental results show using two graphs in Figure 6.7 and Figure 6.8. Monitoring the training and validation accuracy progress plot is helpful when training any network. It shows how quickly accuracy is improving and whether a network is starting to overfit with data or not. The training and validation accuracy plot show that the network is not overfitting or underfitting. Figure 6.7 shows the training and validation accuracy plot versus the number of iterations. This designed network is not having any overfitting issue with data. Initially, at zero iteration, accuracy is low. However, after the 100th iteration, the accuracy increases and becomes constant for a few iterations and at max epoch 500th, it ends up with an accuracy of 88.89%.

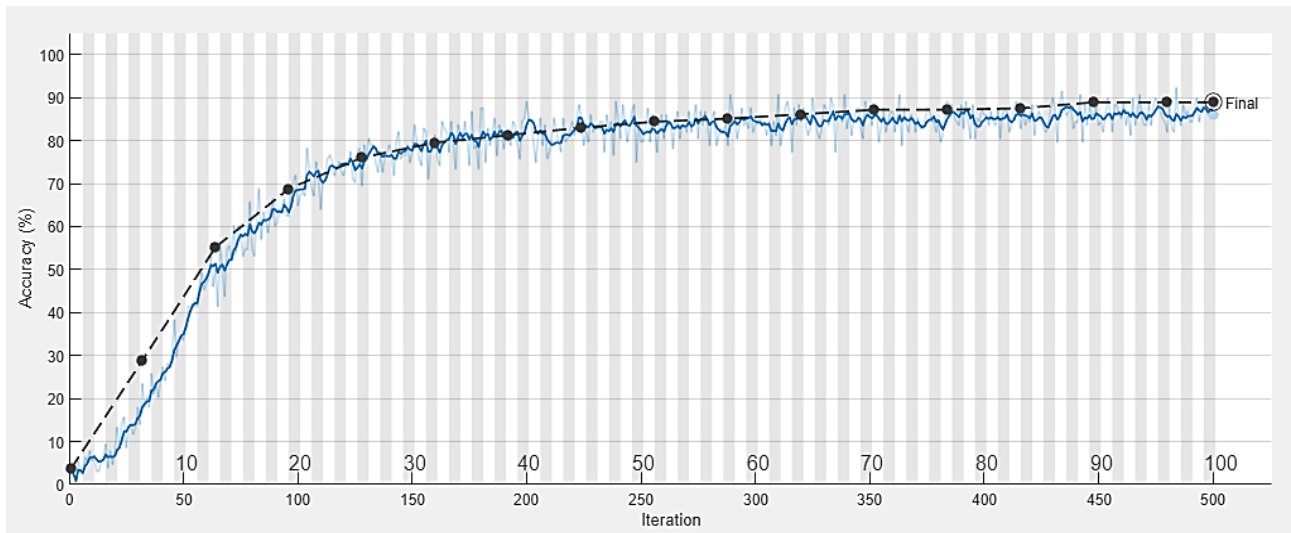


Figure 6.7 Training progress plot (Accuracy versus number of iterations) [Blue line represents training progress and the Black line represents validation progress]

Figure 6.7 shows the training and validation loss function versus the iteration plot. The training and validation losses perfectly fit with the data. However, the loss is more when the batch size is small. As batch size increases, the loss decreases, as given in Figure 6.8. The wiggle is minimal when the batch size is the entire dataset because gradient update improves the loss function monotonically. At the maximum epoch, the final validation loss ends up at 1.45. Table 6.1 compares proposed methodologies with other states of art methods in terms of accuracy. That comparison shows that the proposed method can prefer for action recognition.

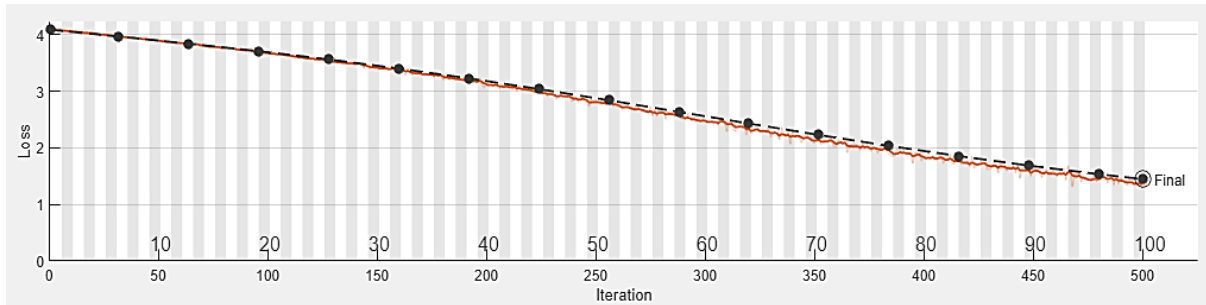


Figure 6.8 Validation loss plot (Loss versus number of iterations) [Red line represents training loss progress and the Black line represents validation loss progress]

6.4 Comparing our approach to other existing methods

The proposed work compares with the methodology given in table 6.1.

Table 6.1 Comparison with other methods based on accuracies

| Method (References) | Accuracy (%) |
|--|--------------|
| Multidimensional Indexing [34] | 84.6 |
| HMM [21] | 88.3 |
| PCA and HMM [29] | 87.5 |
| Memory-based attention control system [16] | 80 |
| Dynamic time warping [31] | 80.05 |
| Bipartite graph [19] | 82.8 |
| Multi-sensor fusion [36] | 88 |
| Deep Learning-based hierarchical feature model [44] | 70.32 |
| Deep convolutional neural network [45] | 41.5 |
| Collaborative Sparse coding [27] | 79.18 |
| Convex multiview semi-supervised classification [60] | 59.08 |
| Scene flow to action map and ConvNets [40] | 61.94 |
| Convolutional neural network [61] | 66.29 |
| Multiview fusion[41] | 85.9 |
| Graph convolutional networks [54] | 88.2 |
| Proposed Method | 88.89 |

6.5 Activity ID with Activity name in dataset (NTURGB+D)

The NURGB+D dataset contain 60 classes allotted with activity ID which are given in table 6.2.

Table 6.2 Activity ID with Activity Name

| Activity_ID | Name of Activity | Activity_ID | Name of Activity | Activity_ID | Name of Activity | Activity_ID | Name of Activity |
|-------------|------------------|-------------|------------------|-------------|------------------|-------------|------------------|
| A001 | drink water | A016 | put on a shoe | A031 | point to | A046 | back pain |

| | | | | | | | |
|------|-----------------|------|---------------------------|------|----------------------------|------|--------------------|
| | | | | | something | | |
| A002 | eat meal | A017 | take off a shoe | A032 | taking a selfie | A047 | neck pain |
| A003 | brush teeth | A018 | put on glasses | A033 | check time (from watch) | A048 | nausea/vomiting |
| A004 | brush hair | A019 | take off glasses | A034 | rub two hands | A049 | fan self |
| A005 | drop | A020 | put on a hat/cap | A035 | nod head/bow | A050 | punch/slap |
| A006 | pick up | A021 | take off a hat/cap | A036 | shake head | A051 | kicking |
| A007 | throw | A022 | cheer up | A037 | wipe face | A052 | pushing |
| A008 | sit down | A023 | hand waving | A038 | Salute | A053 | pat on back |
| A009 | stand up | A024 | kicking something | A039 | put palms together | A054 | point finger |
| A010 | clapping | A025 | reach into pocket | A040 | cross hands in front | A055 | hugging |
| A011 | reading | A026 | hopping | A041 | sneeze/cough | A056 | giving object |
| A012 | writing | A027 | jump up | A042 | Staggering | A057 | touch pocket |
| A013 | tear up paper | A028 | phone call | A043 | falling down | A058 | shaking hands |
| A014 | put on jacket | A029 | play with phone/tablet | A044 | Headache | A059 | walking towards |
| A015 | take off jacket | A030 | type on a keyboard | A045 | chest pain | A060 | walking apart |

6.6 Confusion matrix

Figure 6.9 shows the confusion matrix for 60 classes where each row instance depicts actual classes and each column as predicted classes. The diagonal green colored boxes show the correct number of classes identified. For example, considering the first row and column, the class label with A001 is the same in 6 videos. However, one Video of A001 recognizes as A003. Likewise, the whole matrix can understand.

Table 6.2 shows the class label with the activity name as recognized actions. Each class is labeled with label. For an example, A001 is the activity ID and this class name dedicated to “drinking” activity.

6.7 Two stream model for Polydomain learning

The architecture of two stream network for polydomain learning as shown in figure 6.10. In this network, the input is RGB videos and Skeleton 3D data. The two proposed networks are used as mentioned in chapter 5 and this chapter. The output scores are the output of the two models. Output scores combine with the Late average fusion and consider as feature for next network. The network is design for many type (RGB and Skeleton 3D) and Many view (3 view) inputs. This called as polydomain learning model. The model is train, test and validate to find final action label. The detailed model with layer as shown in figure 6.11. The detail description already explained in chapter 5 and chapter 6.

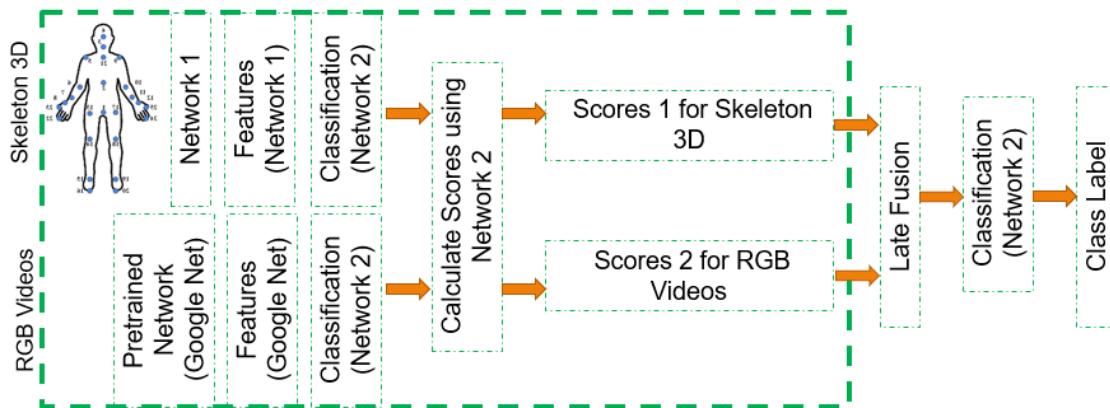


Figure 6.10 Two stream networks of polydomain learning of action recognition

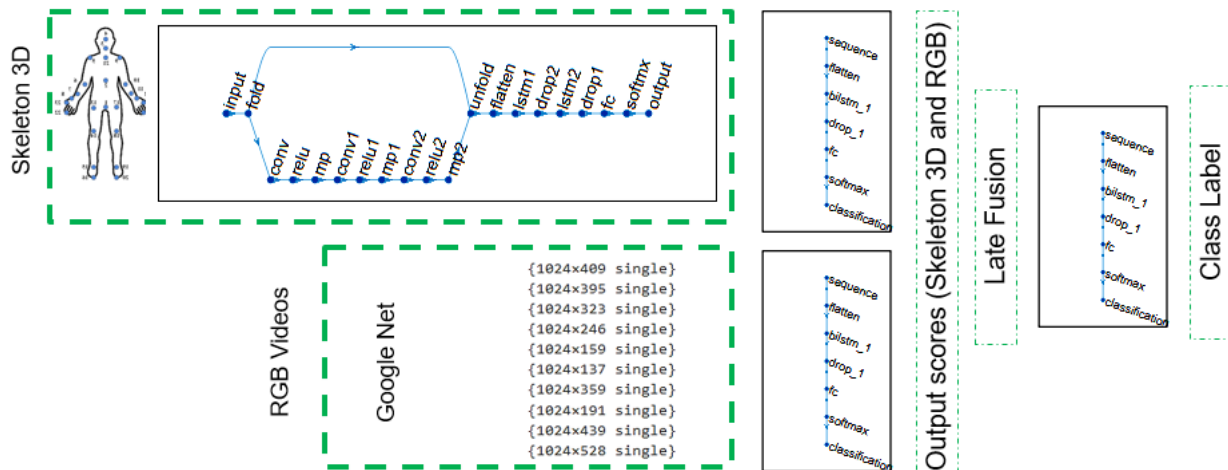


Figure 6.11 Detailed two stream networks of polydomain learning of action recognition

6.7.1 Experimental analysis

The experiment evaluation was performed on the NTURGB+D dataset. The number of videos uses for performance evaluation is 960 in number. The variation of accuracy with iteration shows in figure 6.12. The accuracy which starts from 0th and near the 50th iteration reaches at 67% and then ends at 85.76%. The graph shown in figure 6.13, where the validation loss with iteration shows and ends at 1.76.

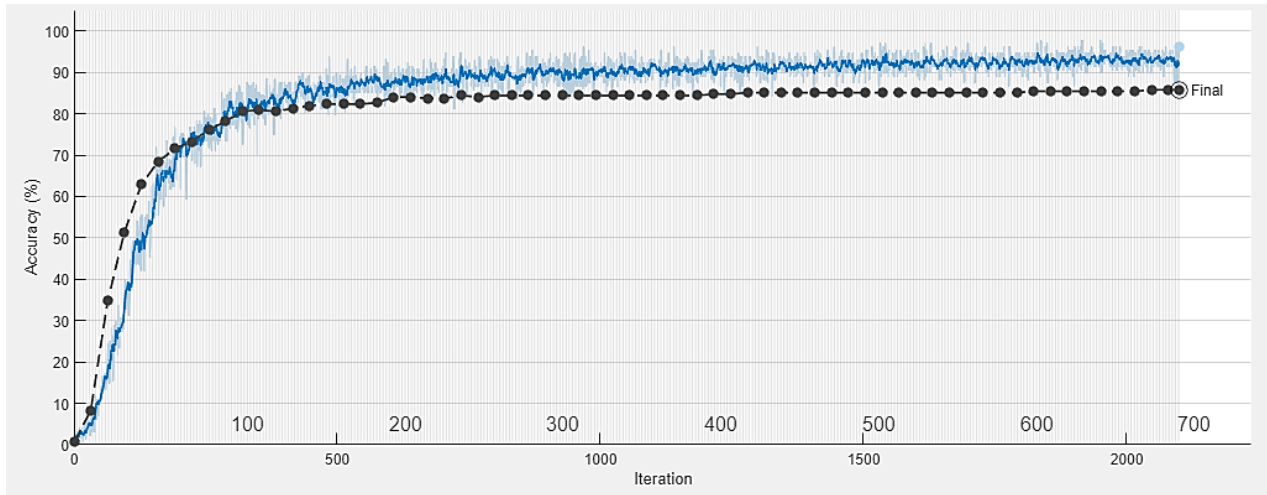


Figure 6.12 Variation of Accuracy vs number of iterations

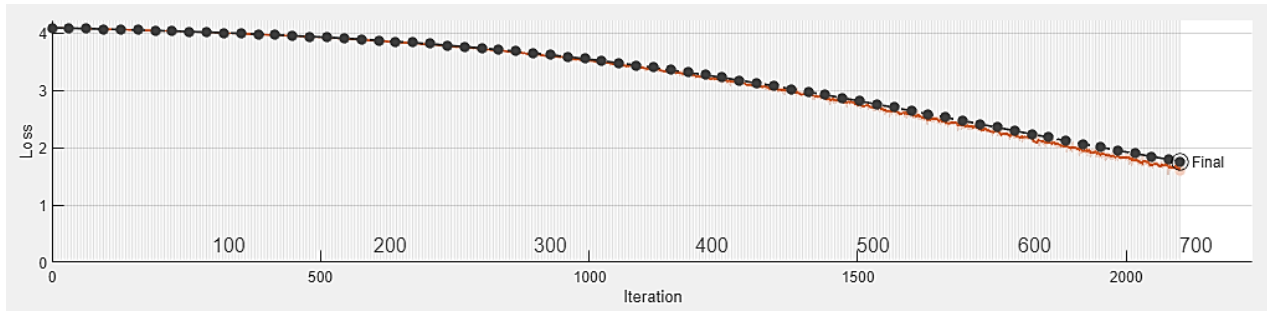


Figure 6.13 Variation of Loss versus number of iterations

6.8 Summary

This work presents multiview-based skeleton action recognition using deep neural networks. This work proposes the networks, i.e., CNN_RNN_1 and CNN_RNN_2, where CNN_RNN_1 uses for feature reduction technique and CNN_RNN_2 for action classification. The activities classification uses 3D skeleton information of all three views from the dataset NTURGB+D for 60 classes. The designed system outperforms all the other state-of-the-art

methods. The accuracy of action can improve by including more layers in the network.

The architecture of two stream networks proposed by using designed networks CNN_RNN_1 and CNN_RNN_2 networks. The NTURGB+D dataset uses to design action recognition model. The overall accuracy given by the model is 85.76%. In future work, the model's accuracy can improve for more classes.

The system can also design with three-stream input networks to improve evaluation parameters. This work will extend by developing three-stream networks.

CHAPTER-7

CONCLUSIONS AND FUTURE DIRECTION

7.1 Summary of contributions

The HAR plays a vital role in the field of surveillance and monitoring. This research contributes to designing a system that can recognize activities. So that timely action can take for the safety of people. There are three significant tasks on which research is carried out i.e. extraction of features, reduction of features and system which can recognize activities.

In Chapter 3, the proposed work gave an approach to segment all classes of RGB video using skeletal 3D information. This dataset is one of the biggest of its kind and employs NTURGB+D with 4 modalities: RGB, a 3D skeleton, a depth map, and an IR sequence. It is especially beneficial for approaches which require a large number of classes or views. The size of the segmentation window is adaptively chosen based on the RGB video using skeletal 3D information. This technique works well for all NTURGB+D videos. In the Experimental Results section, some examples of random video results are given. It is possible to observe the sectioned output video. Generally, all videos perform satisfactorily, apart from interactive activities. This window requires explicit modification if categories of interaction are to be considered. Based on experiment, the segmented percentage evaluated. Its observed that, the maximum segmented percentage is 95.48%. This way of segmentation can be used for any experiment. Future work will improve this technique when there are multiple people in the video.

In Chapter 3, this study shows how object recognition can be achieved through semantic segmentation. A Deep La v3+ network using resnet18 was employed to segment images from the CamVid dataset for scene classification into 11 classes. There are three metrics which are accuracy, IoU and mean scores is used for experiment analysis. The accuracy, IoU, and score of the model were tested and the highest accuracy per class was reported in the results section. The highest accuracy, at 0.9505, is achieved in the "Sky" class, where the area in the image is more prominent. Conversely, for smaller objects such as "Pole," "SignSymbol," and "Fence," the accuracy decreases, reflecting the model's challenges in recognizing these smaller elements. Remarkably,

the model exhibits strong performance in identifying larger objects. In terms of Intersection over Union (IoU), the "Pole" class attains a minimal value of 0.2629 due to overlaps between the predicted and actual bounding boxes. In contrast, the "Sky" class reaches the highest IoU score at 0.9088. The mean scores provide an overview of how accurately the model classifies instances. Among the classes, "Sky" achieves the highest mean score of 0.9044, while "SignSymbol" obtains the lowest at 0.5286. This technique had challenges, such as difficulty in recognizing small objects in the frame, but it worked well with large objects.

Semantic segmentation can be used in a variety of applications, including self-driving cars, behavioural recognition, and healthcare. One of the most important first steps in discovery or discovery is finding available objects. This model can be used in the future by focusing on small objects in images/videos. This model is a good model for semantic segmentation if smaller objects are also segmented accordingly.

The experimental analysis for motion estimation from videos for understanding motions can be challenging in chapter 4. In this analysis, optical flow visualization is analysed using four methods. Each technique helps calculate two main parameters: magnitude and orientation. The magnitude gives an understanding of brightness variation from pixel to pixel, and orientation gives the movement of directions. The optical flow visualization shows one sample video. It analyzed using PSNR that variation between two video frames is the same in the optical flow magnitude of each method. The PSNR was calculated for 15 videos, from which the PSNR for Lucas Kanade often comes higher. The more the value of PSNR, the better the quality of an image. Based on the visualization and PSNR, the motion estimating optical flow using Lucas Kanade outperforms the other three motion estimators. The Lucas Kanade can be the preference over others for applications like action recognition, 3D reconstructions, or video coding. The work can extend to using these features for motion estimations.

In Chapter 5, a BiLSTM model with transfer learning is explored. This model is on par with other existing methods. However, they all suffer from certain limitations, one of which is the accuracy rate. When the model was tested with 10 classes, it achieved a very high recognition rate. On the other hand, when the system was tested with 51 classes, the accuracy dropped to 63.96%. Thus, the accuracy of the model depends on many factors, including the feature vector,

hyperparameters, and number of layers. Of these, the most vital concept is the tuning of the model's hyperparameters. This is highly significant for models based on deep learning.

In our next steps, we can set the hyperparameters and construct a system based on supervised learning, using multiple training videos. This model is based on a single-view dataset, but can be extended to create an optimal model suitable for data sets with multiple views.

In chapter 6, the proposed work presents multiview-based skeleton action recognition using deep neural networks. We proposed the networks, i.e., CNN_RNN_1 and CNN_RNN_2, where CNN_RNN_1 uses for feature reduction technique and CNN_RNN_2 for action classification. The activities classification uses 3D skeleton information of all three views from the dataset NTURGB+D for 60 classes.

A two-stream network architecture is introduced, incorporating the specially designed CNN_RNN_1 and CNN_RNN_2 networks. These networks are employed in developing an action recognition model using the NTURGB+D dataset, which achieves an overall accuracy of 85.76%. Future enhancements are anticipated to further enhance accuracy, particularly for additional classes.

Additionally, this can expand the system's capabilities by incorporating three-stream input networks to enhance evaluation parameters. This endeavor will involve the development of three-stream networks to extend the research work.

7.2 Challenges and Future Directions

The HAR is the most thrust research area in video signal processing. No doubt, there are various methodologies have been proposed by researchers. However, there are still open challenges in which directions require work. The available challenges are data collection, processing, complex activities, misalignment of activities in the videos, hardware limitations, and the design of deep learning hybrid models.

Misalignment of activities: Manual annotation of data is a time-consuming task. Sometimes, the number of frames is missing, and joint information is unavailable, or incorrect labeling. Dataset can use, which is preferred mainly by researchers. In this work, a few samples are unavailable for

skeleton or RGB video frames in the NTURGB+D.

Hardware limitation: For deep learning-based models, which are called data-hungry techniques. There is a requirement for the high-speed computational power of the system. Multi-GPU parallel processing is the preference for quick and fast activity recognition for deep learning-based models. In this thesis, the experiment performs on the system with a 1.6 GHz Intel Core i5-4200U, 8GB RAM, and 1TB SSD running a Windows 10 with the 64-bit operating system. The model can be trained, validated, and tested more effectively if the system is running at a high speed, allowing for more videos to be used.

Design of deep learning hybrid models: Hybrid deep learning models can design for a more significant design of HAR. The model can take multiple modalities and views for an efficient model for HAR. In this thesis, the two-stream network presents recognition actions. For more recognition rate, we can design more stream networks also.

The influence of recognition systems on a range of motivational application areas, including behavioural biometrics, content-based video analytics, security and surveillance, interactive applications and environments, animation and synthesis, is becoming increasingly significant.

Lastly, there is always a chance to improve existing systems based on research carried out by the researcher. With this purpose, this research will carry out in future for further on.

REFERENCES

- [1] J. K. Aggarwal and M. S. Ryoo, “Human Activity Analysis : A Review,” *ACM Computing Surveys*, vol. 43, no. 3, pp. 1–43, Apr. 2011.
- [2] G. Johansson, “Visual perception of biological motion and a model for its analysis,” *Perception & Psychophysics*, vol. 14, no. 2, pp. 201–211, Jun. 1973.
- [3] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea, “Machine Recognition of Human Activities: A Survey,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1473–1488, Nov. 2008.
- [4] S. Sarkar, P. J. Phillips, Z. Liu, I. R. Vega, P. Grother, and K. W. Bowyer, “The humanID gait challenge problem: data sets, performance, and analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 2, pp. 162–177, Feb. 2005.
- [5] M. Ramezani and F. Yaghmaee, “A review on human action analysis in videos for retrieval applications,” *Artificial Intelligence Review*, vol. 46, no. 4, pp. 485–514, Dec. 2016.
- [6] C. Stauffer and W. E. L. Grimson, “Learning patterns of activity using real-time tracking,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 747–757, 2000.
- [7] O. P. Popoola and Kejun Wang, “Video-Based Abnormal Human Behavior Recognition—A Review,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 6, pp. 865–878, Nov. 2012.
- [8] G. Sannino, I. De Falco, and G. De Pietro, “A supervised approach to automatically extract a set of rules to support fall detection in an mHealth system,” *Applied Soft Computing Journal*, vol. 34, pp. 205–216, 2015.
- [9] N. Lu, Y. Wu, L. Feng, and J. Song, “Deep Learning for Fall Detection: 3D-CNN Combined with LSTM on Video Kinematic Data,” *IEEE Journal of Biomedical and Health Informatics*, vol. 2194, no. c, pp. 1–1, 2018.
- [10] A.-A. Liu, N. Xu, W.-Z. Nie, Y.-T. Su, Y. Wong, and M. Kankanhalli, “Benchmarking a Multimodal and Multiview and Interactive Dataset for Human Action Recognition,” *IEEE Transactions on Cybernetics*, vol. 47, no. 7, pp. 1781–1794, Jul. 2017.
- [11] Z. Gao, H. Zhang, G. P. Xu, Y. B. Xue, and A. G. Hauptmann, “Multi-view discriminative and structured dictionary learning with group sparsity for human action recognition,” *Signal*

- Processing*, vol. 112, pp. 83–97, Jul. 2015.
- [12] J. Zheng, Z. Jiang, and R. Chellappa, “Cross-View Action Recognition via Transferable Dictionary Learning,” *IEEE Transactions on Image Processing*, vol. 25, no. 6, pp. 2542–2556, Jun. 2016.
- [13] T. Singh and D. K. Vishwakarma, “Human Activity Recognition in Video Benchmarks: A Survey,” vol. 526, B. S. Rawat, A. Trivedi, S. Manhas, and V. Karwal, Eds. Singapore: Springer Singapore, 2019, pp. 247–259.
- [14] L. Alzubaidi *et al.*, “Review of deep learning: concepts, CNN architectures, challenges, applications, future directions,” *Journal of Big Data*, vol. 8, no. 1, p. 53, Dec. 2021.
- [15] Y. Liu, Z. Lu, J. Li, T. Yang, and C. Yao, “Global Temporal Representation Based CNNs for Infrared Action Recognition,” *IEEE Signal Processing Letters*, vol. 25, no. 6, pp. 848–852, Jun. 2018.
- [16] K. F. MacDorman, H. Nobuta, S. Koizumi, and H. Ishiguro, “Memory-Based Attention Control for Activity Recognition at a Subway Station,” *IEEE Multimedia*, vol. 14, no. 2, pp. 38–49, Apr. 2007.
- [17] M. Singh, A. Basu, and M. K. Mandal, “Human Activity Recognition Based on Silhouette Directionality,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 9, pp. 1280–1292, Sep. 2008.
- [18] M. B. Holte, C. Tran, M. M. Trivedi, and T. B. Moeslund, “Human Pose Estimation and Activity Recognition From Multi-View Videos: Comparative Explorations of Recent Developments,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, no. 5, pp. 538–552, Sep. 2012.
- [19] J. Liu, M. Shah, B. Kuipers, and S. Savarese, “Cross-view action recognition via view knowledge transfer,” in *CVPR 2011*, 2011, pp. 3209–3216.
- [20] B. Liang and L. Zheng, “Specificity and Latent Correlation Learning for Action Recognition Using Synthetic Multi-View Data From Depth Maps,” *IEEE Transactions on Image Processing*, vol. 26, no. 12, pp. 5560–5574, Dec. 2017.
- [21] Feng Niu and M. Abdel-Mottaleb, “View-Invariant Human Activity Recognition Based on Shape and Motion Features,” in *IEEE Sixth International Symposium on Multimedia Software Engineering*, 2004, pp. 546–556.
- [22] M. Zia Uddin, J. J. Lee, and T.-S. Kim, “Independent shape component-based human

- activity recognition via Hidden Markov Model,” *Applied Intelligence*, vol. 33, no. 2, pp. 193–206, Oct. 2010.
- [23] A. Iosifidis, A. Tefas, and I. Pitas, “View-Invariant Action Recognition Based on Artificial Neural Networks,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 3, pp. 412–424, Mar. 2012.
- [24] L. Maddalena and A. Petrosino, “A Self-Organizing Approach to Background Subtraction for Visual Surveillance Applications,” *IEEE Transactions on Image Processing*, vol. 17, no. 7, pp. 1168–1177, Jul. 2008.
- [25] A. Iosifidis, A. Tefas, and I. Pitas, “Multi-view action recognition based on action volumes, fuzzy distances and cluster discriminant analysis,” *Signal Processing*, vol. 93, no. 6, pp. 1445–1457, Jun. 2013.
- [26] N. Gkalelis, N. Nikolaidis, and I. Pitas, “View independent human movement recognition from multi-view video exploiting a circular invariant posture representation,” in *2009 IEEE International Conference on Multimedia and Expo*, 2009, pp. 394–397.
- [27] W. Wang, Y. Yan, L. Zhang, R. Hong, and N. Sebe, “Collaborative Sparse Coding for Multiview Action Recognition,” *IEEE MultiMedia*, vol. 23, no. 4, pp. 80–87, Oct. 2016.
- [28] A. A. Chaaoui, P. Climent-Pérez, and F. Flórez-Revuelta, “Silhouette-based human action recognition using sequences of key poses,” *Pattern Recognition Letters*, vol. 34, no. 15, pp. 1799–1807, Nov. 2013.
- [29] M. Ahmad and Seong-Whan Lee, “HMM-based Human Action Recognition Using Multiview Image Sequences,” in *18th International Conference on Pattern Recognition (ICPR’06)*, 2006, vol. 1, pp. 263–266.
- [30] M. Ahmad and S.-W. Lee, “Human action recognition using shape and CLG-motion flow from multi-view image sequences,” *Pattern Recognition*, vol. 41, no. 7, pp. 2237–2252, Jul. 2008.
- [31] S. Cherla, K. Kulkarni, A. Kale, and V. Ramasubramanian, “Towards fast, view-invariant human action recognition,” in *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2008, pp. 1–8.
- [32] Jingen Liu, S. Ali, and M. Shah, “Recognizing human actions using multiple features,” in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.
- [33] K. K. Reddy, J. Liu, and M. Shah, “Incremental action recognition using feature-tree,”

- Proceedings of the IEEE International Conference on Computer Vision*, no. Iccv, pp. 1010–1017, 2009.
- [34] J. Ben-Arie, Zhiqian Wang, P. Pandit, and S. Rajaram, “Human activity recognition using multidimensional indexing,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 8, pp. 1091–1104, Aug. 2002.
- [35] Yale Song, L. Morency, and R. Davis, “Multi-view latent variable discriminative models for action recognition,” in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2120–2127.
- [36] F. Zhu, L. Shao, and M. Lin, “Multi-view action recognition using local similarity random forests and sensor fusion,” *Pattern Recognition Letters*, vol. 34, no. 1, pp. 20–24, Jan. 2013.
- [37] A.-A. Liu, Y.-T. Su, P.-P. Jia, Z. Gao, T. Hao, and Z.-X. Yang, “Multiple/Single-View Human Action Recognition via Part-Induced Multitask Structural Learning,” *IEEE Transactions on Cybernetics*, vol. 45, no. 6, pp. 1194–1208, Jun. 2015.
- [38] Y. Guo, D. Tao, W. Liu, and J. Cheng, “Multiview Cauchy Estimator Feature Embedding for Depth and Inertial Sensor-Based Human Action Recognition,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 4, pp. 617–627, Apr. 2017.
- [39] M. A. Naiel, M. M. Abdelwahab, and M. El-Saban, “Multi-view human action recognition system employing 2DPCA,” in *2011 IEEE Workshop on Applications of Computer Vision (WACV)*, 2011, pp. 270–275.
- [40] P. Wang, W. Li, Z. Gao, Y. Zhang, C. Tang, and P. Ogunbona, “Scene flow to action map: A new representation for RGB-D based action recognition with convolutional neural networks,” *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 416–425, 2017.
- [41] Z. Fan, X. Zhao, T. Lin, and H. Su, “Attention-Based Multiview Re-Observation Fusion Network for Skeletal Action Recognition,” *IEEE Transactions on Multimedia*, vol. 21, no. 2, pp. 363–374, Feb. 2019.
- [42] W. Yang, Y. Shi, Y. Gao, L. Wang, and M. Yang, “Incomplete-Data Oriented Multiview Dimension Reduction via Sparse Low-Rank Representation,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 12, pp. 6276–6291, Dec. 2018.
- [43] 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*, vol. 38, no. 10, pp. 2123–2129, Oct. 2016.
- [44] M. Hasan and A. K. Roy-Chowdhury, “A Continuous Learning Framework for Activity Recognition Using Deep Hybrid Feature Models,” *IEEE Transactions on Multimedia*, vol. 17, no. 11, pp. 1909–1922, Nov. 2015.
- [45] H. Zhu, J.-B. Weibel, and S. Lu, “Discriminative Multi-modal Feature Fusion for RGBD Indoor Scene Recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2969–2976.
- [46] P. Wang, W. Li, Z. Gao, J. Zhang, C. Tang, and P. O. Ogunbona, “Action Recognition from Depth Maps Using Deep Convolutional Neural Networks,” *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 4, pp. 498–509, 2016.
- [47] A. Shahroudy, T. Ng, Y. Gong, and G. Wang, “Deep Multimodal Feature Analysis for Action Recognition in RGB+D Videos,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 5, pp. 1045–1058, May 2018.
- [48] H. El-Ghaish, M. E. Hussein, A. Shoukry, and R. Onai, “Human Action Recognition Based on Integrating Body Pose, Part Shape, and Motion,” *IEEE Access*, vol. 6, no. c, pp. 49040–49055, 2018.
- [49] C. Li, Y. Hou, P. Wang, and W. Li, “Multiview-Based 3-D Action Recognition Using Deep Networks,” *IEEE Transactions on Human-Machine Systems*, vol. 49, no. 1, pp. 95–104, Feb. 2019.
- [50] D. Weinland, R. Ronfard, and E. Boyer, “Free viewpoint action recognition using motion history volumes,” *Computer Vision and Image Understanding*, vol. 104, no. 2-3 SPEC. ISS., pp. 249–257, 2006.
- [51] N. Gkalelis, H. Kim, A. Hilton, N. Nikolaidis, and I. Pitas, “The i3DPost Multi-View and 3D Human Action/Interaction Database,” in *2009 Conference for Visual Media Production*, 2009, pp. 159–168.
- [52] S. Singh, S. A. Velastin, and H. Ragheb, “MuHAVi: A Multicamera Human Action Video Dataset for the Evaluation of Action Recognition Methods,” in *2010 7th IEEE International Conference on Advanced Video and Signal Based Surveillance*, 2010, pp. 48–55.
- [53] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, “NTU RGB+D: A Large Scale Dataset for 3D Human Activity Analysis,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1010–1019.

- [54] J. Xie *et al.*, “Cross-Channel Graph Convolutional Networks for Skeleton-Based Action Recognition,” *IEEE Access*, vol. 9, pp. 9055–9065, 2021.
- [55] H. Zhang and L. E. Parker, “CoDe4D: Color-Depth Local Spatio-Temporal Features for Human Activity Recognition From RGB-D Videos,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 3, pp. 541–555, Mar. 2016.
- [56] B. Ayhan, C. Kwan, B. Budavari, J. Larkin, D. Gribben, and B. Li, “Video Activity Recognition With Varying Rhythms,” *IEEE Access*, vol. 8, pp. 191997–192008, 2020.
- [57] J. Wang, J. Hu, S. Li, and Z. Yuan, “Revisiting Hard Example for Action Recognition,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8215, no. c, pp. 1–1, 2020.
- [58] N. Kase, M. Babae, and G. Rigoll, “Multi-view human activity recognition using motion frequency,” in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 3963–3967.
- [59] X. Wu and Y. Jia, “View-Invariant Action Recognition Using Latent Kernelized Structural SVM,” 2012, pp. 411–424.
- [60] F. Nie, J. Li, and X. Li, “Convex Multiview Semi-Supervised Classification,” *IEEE Transactions on Image Processing*, vol. 26, no. 12, pp. 5718–5729, Dec. 2017.
- [61] A.-A. Liu, N. Xu, W.-Z. Nie, Y.-T. Su, and Y.-D. Zhang, “Multi-Domain and Multi-Task Learning for Human Action Recognition,” *IEEE Transactions on Image Processing*, vol. 28, no. 2, pp. 853–867, Feb. 2019.
- [62] J. Guo, Z. Li, L. F. Cheong, and S. Z. Zhou, “Video co-segmentation for meaningful action extraction,” *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2232–2239, 2013.
- [63] H. Jain and G. Harit, “Unsupervised Temporal Segmentation of Human Action Using Community Detection,” *Proceedings - International Conference on Image Processing, ICIP*, pp. 1892–1896, 2018.
- [64] G.-J. Chen, I.-C. Chang, and H.-Y. Yeh, “Action segmentation based on Bag-of-Visual-Words models,” in *2017 10th International Conference on Ubi-media Computing and Workshops (Ubi-Media)*, 2017, pp. 1–5.
- [65] Q. Shi, L. Wang, L. Cheng, and A. Smola, “Discriminative Human Action Segmentation and Recognition using {SMM}s,” *International Journal on Computer Vision*, vol. 93, no.

- 1, pp. 22–32, 2011.
- [66] F. Lv and R. Nevatia, “Recognition and segmentation of 3-D human action using HMM and multi-class AdaBoost,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3954 LNCS, pp. 359–372, 2006.
- [67] W. Luo *et al.*, “Action Unit Memory Network for Weakly Supervised Temporal Action Localization,” in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 9964–9974.
- [68] Z. Li, Y. A. Farha, and J. Gall, “Temporal Action Segmentation from Timestamp Supervision,” in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 8361–8370.
- [69] S.-H. Gao, Q. Han, Z.-Y. Li, P. Peng, L. Wang, and M.-M. Cheng, “Global2Local: Efficient Structure Search for Video Action Segmentation,” in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 16800–16809.
- [70] D. Gong, G. Medioni, and X. Zhao, “Structured Time Series Analysis for Human Action Segmentation and Recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1414–1427, Jul. 2014.
- [71] S. Park, U. Park, and D. Kim, “Depth image-based object segmentation scheme for improving human action recognition,” *International Conference on Electronics, Information and Communication, ICEIC 2018*, vol. 2018-Janua, pp. 1–3, 2018.
- [72] F. Murtaza, M. H. Yousaf, and S. A. Velastin, “PMHI: Proposals from motion history images for temporal segmentation of long uncut videos,” *IEEE Signal Processing Letters*, vol. 25, no. 2, pp. 179–183, 2018.
- [73] Y. Han, S. L. Chung, and S. F. Su, “Automatic action segmentation and continuous recognition for basic indoor actions based on kinect pose streams,” *2017 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2017*, vol. 2017-Janua, pp. 966–971, 2017.
- [74] G. Tanışık, Öguzhan Güçlü, and N. İkizler-Cinbis, “Bölüt ve kontur özneliklerini kullanarak imgelerdeki insan hareketlerini tanıma,” *2013 21st Signal Processing and Communications Applications Conference, SIU 2013*, 2013.
- [75] A. Ess, T. Müller, H. Grabner, and L. Van Gool, “Segmentation-based urban traffic scene

- understanding,” *British Machine Vision Conference, BMVC 2009 - Proceedings*, pp. 1–11, 2009.
- [76] Y. Yoon, H. G. Jeon, D. Yoo, J. Y. Lee, and I. S. Kweon, “Learning a Deep Convolutional Network for Light-Field Image Super-Resolution,” *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015-Febru, pp. 57–65, 2015.
- [77] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, P. Martinez-Gonzalez, and J. Garcia-Rodriguez, “A survey on deep learning techniques for image and video semantic segmentation,” *Applied Soft Computing Journal*, vol. 70, pp. 41–65, 2018.
- [78] H. T. Ong and K. K. Ma, “Semantic image segmentation using oriented pattern analysis,” *ICICS 2011 - 8th International Conference on Information, Communications and Signal Processing*, pp. 1–4, 2011.
- [79] K. Luo, F. Meng, Q. Wu, and H. Li, “Weakly Supervised Semantic Segmentation by Multiple Group Cosegmentation,” *VCIP 2018 - IEEE International Conference on Visual Communications and Image Processing*, pp. 1–4, 2018.
- [80] M. Zhang, S. Liu, Y. Zhu, Z. Bai, and J. Lin, “SEG-HASHNET: SEMANTIC SEGMENTATION BASED UNSUPERVISED HASHING Institute of Big Data Technology , Shenzhen Graduate School , Peking University School of Software & Microelectronics , Peking University,” pp. 12–16.
- [81] H. Lyu, H. Fu, X. Hu, and L. Liu, “Esnet: Edge-Based Segmentation Network for Real-Time Semantic Segmentation in Traffic Scenes,” *Proceedings - International Conference on Image Processing, ICIP*, vol. 2019-Septe, pp. 1855–1859, 2019.
- [82] Q. Zheng, J. Chen, P. Huang, and R. Hu, “Urban scene semantic segmentation with insufficient labeled data,” *China Communications*, vol. 16, no. 11, pp. 212–221, 2019.
- [83] Z. Guo *et al.*, “Semantic segmentation for urban planning maps based on U-Net,” *International Geoscience and Remote Sensing Symposium (IGARSS)*, vol. 2018-July, pp. 6187–6190, 2018.
- [84] N. Noormohamadi, P. Adibi, and S. M. S. Ehsani, “Semantic image segmentation using an improved hierarchical graphical model,” *IET Image Processing*, vol. 12, no. 11, pp. 1943–1950, 2018.
- [85] M. Arunkumar and V. Pushparaj, “Seed picking crossover optimisation algorithm for semantic segmentation from images,” *IET Image Processing*, vol. 14, no. 11, pp. 2503–

- 2511, Sep. 2020.
- [86] X. Chen, A. Wu, and Y. Han, "Capturing the spatio-temporal continuity for video semantic segmentation," *IET Image Processing*, vol. 13, no. 14, pp. 2813–2820, 2019.
 - [87] L. Huang, M. He, C. Tan, J. Du, G. Li, and H. Yu, "Jointly network image processing: Multi-task image semantic segmentation of indoor scene based on CNN," *IET Image Processing*, vol. 14, no. 15, pp. 3689–3697, 2020.
 - [88] H. Li, X. Qian, and W. Li, "Image Semantic Segmentation Based on Fully Convolutional Neural Network and CRF," in *Communications in Computer and Information Science*, vol. 698, H. Yuan, J. Geng, and F. Bian, Eds. Singapore: Springer Singapore, 2017, pp. 245–250.
 - [89] J. An, H. Zhang, Y. Zhu, and J. Yang, "Semantic Segmentation for Prohibited Items in Baggage Inspection," vol. 2, 2019, pp. 495–505.
 - [90] G. Li, L. Li, and J. Zhang, "BiAttnNet: Bilateral Attention for Improving Real-time Semantic Segmentation," *IEEE Signal Processing Letters*, pp. 1–1, 2021.
 - [91] A. Syed and B. T. Morris, "SSeg-LSTM: Semantic Scene Segmentation for Trajectory Prediction - IEEE Conference Publication," *2019 IEEE Intelligent Vehicles Symposium (IV)*, no. Iv, pp. 2504–2509, 2019.
 - [92] L. C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11211 LNCS, pp. 833–851, 2018.
 - [93] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88–97, 2009.
 - [94] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-Decem, pp. 770–778, 2016.
 - [95] A. Ahmed, K. Yu, W. Xu, Y. Gong, and E. Xing, "Training hierarchical feed-forward visual recognition models using transfer learning from Pseudo-tasks," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5304 LNCS, no. PART 3, pp. 69–82, 2008.

- [96] T. B. Moeslund, A. Hilton, and V. Krüger, “A survey of advances in vision-based human motion capture and analysis,” *Computer Vision and Image Understanding*, vol. 104, no. 2-3 SPEC. ISS., pp. 90–126, 2006.
- [97] U. Mahbub, H. Imtiaz, and M. A. Rahman Ahad, “An optical flow based approach for action recognition,” in *14th International Conference on Computer and Information Technology (ICCIT 2011)*, 2011, pp. 646–651.
- [98] M. Lucena, N. P. De La Blanca, J. M. Fuertes, and M. J. Marín-Jiménez, “Human action recognition using optical flow accumulated local histograms,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5524 LNCS, pp. 32–39, 2009.
- [99] A. Gupta and M. S. Balan, *Action recognition from optical flow visualizations*, vol. 703. Springer Singapore, 2018.
- [100] J. K. Aggarwal and Q. Cai, “Q. Cai, Human Motion Analysis A Review-1,” vol. 73, no. 3, pp. 428–440, 1999.
- [101] T. Brox, A. Bruhn, N. Papenber, and J. Weickert, “High Accuracy Optical Flow Estimation Based on a Theory for Warping,” 2004, pp. 25–36.
- [102] G.-P. Ji, D.-P. Fan, K. Fu, Z. Wu, J. Shen, and L. Shao, “Full-duplex strategy for video object segmentation,” *Computational Visual Media*, vol. 9, no. 1, pp. 155–175, Mar. 2023.
- [103] Y. Bouafia, L. Guezouli, and H. Lakhlef, “Human Detection in Surveillance Videos Based on Fine-Tuned MobileNetV2 for Effective Human Classification,” *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, vol. 46, no. 4, pp. 971–988, Dec. 2022.
- [104] G. Farneb, “Two-Frame Motion Estimation Based on Polynomial Expansion,” *Lecture Notes in Computer Science*, vol. 2749, no. 1, pp. 363–370, 2003.
- [105] B. K. P. Horn and B. G. Schunck, “Determining optical flow,” *Artificial Intelligence*, vol. 17, no. 1–3, pp. 185–203, 1981.
- [106] J. L. Barron, D. J. Fleet, and S. S. Beauchemin, “Performance of optical flow techniques,” *International Journal of Computer Vision*, vol. 12, no. 1, pp. 43–77, Feb. 1994.
- [107] J. L. Barron, D. J. Fleet, S. S. Beauchemin, and T. A. Burkitt, “Performance of optical flow techniques,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1992-June, pp. 236–242, 1992.

- [108] N. Dalal and B. Triggs, “Histograms of Oriented Gradients for Human Detection,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, 2005, vol. 1, pp. 886–893.
- [109] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
- [110] G. Van Houdt, C. Mosquera, and G. Nápoles, “A review on the long short-term memory model,” *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5929–5955, 2020.
- [111] X. Wu and Q. Ji, “TBRNet: Two-stream BiLSTM residual network for video action recognition,” *Algorithms*, vol. 13, no. 7, pp. 1–21, 2020.
- [112] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, “HMDB: A large video database for human motion recognition,” *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2556–2563, 2011.
- [113] Y. Zhu, Y. Long, Y. Guan, S. Newsam, and L. Shao, “Towards Universal Representation for Unseen Action Recognition,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 9436–9445, 2018.
- [114] D. Tran, J. Ray, Z. Shou, S.-F. Chang, and M. Paluri, “ConvNet Architecture Search for Spatiotemporal Feature Learning,” no. section 3, 2017.
- [115] K. Simonyan and A. Zisserman, “Two-stream convolutional networks for action recognition in videos,” *Advances in Neural Information Processing Systems*, vol. 1, no. January, pp. 568–576, 2014.
- [116] N. Srivastava, E. Mansimov, and R. Salakhutdinov, “Unsupervised learning of video representations using LSTMs,” *32nd International Conference on Machine Learning, ICML 2015*, vol. 1, pp. 843–852, 2015.
- [117] L. Sun, K. Jia, D. Y. Yeung, and B. E. Shi, “Human action recognition using factorized spatio-temporal convolutional networks,” *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015 Inter, pp. 4597–4605, 2015.
- [118] H. Wang, C. Schmid, A. Recognition, and T. Iccv, “Action Recognition with Improved Trajectories To cite this version : Action Recognition with Improved Trajectories,” 2013.
- [119] H. Bilen, B. Fernando, E. Gavves, and A. Vedaldi, “Action Recognition with Dynamic Image Networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 12, pp. 2799–2813, 2018.

- [120] T. Pan, Y. Song, T. Yang, W. Jiang, and W. Liu, “VideoMoCo: Contrastive Video Representation Learning with Temporally Adversarial Examples,” 2021.
- [121] B. V. Rani and P. Singh, “A Survey On Electronic Health Records (EHRS): Challenges And Solutions,” in *2022 6th International Conference on Computing Methodologies and Communication (ICCMC)*, 2022, pp. 655–658.
- [122] A. Ahmed, M. M. Khan, P. Singh, R. S. Bath, and M. Masud, “IoT-based real-time patients vital physiological parameters monitoring system using smart wearable sensors,” *Neural Computing and Applications*, Apr. 2022.
- [123] A. Kaur, P. Singh, and A. Nayyar, “Fog Computing: Building a Road to IoT with Fog Analytics,” *Studies in Big Data*, vol. 76, no. August, pp. 59–78, 2020.
- [124] P. Singh, A. Kaur, and N. Kumar, “A reliable and cost-efficient code dissemination scheme for smart sensing devices with mobile vehicles in smart cities,” *Sustainable Cities and Society*, vol. 62, p. 102374, Nov. 2020.
- [125] F. R. Khan *et al.*, “A Cost-Efficient Autonomous Air Defense System for National Security,” *Security and Communication Networks*, vol. 2021, pp. 1–10, Jun. 2021.
- [126] S. Sun, “A survey of multi-view machine learning,” *Neural Computing and Applications*, vol. 23, no. 7–8, pp. 2031–2038, Dec. 2013.
- [127] C. Hong, J. Yu, J. Wan, D. Tao, and M. Wang, “Multimodal Deep Autoencoder for Human Pose Recovery,” *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5659–5670, Dec. 2015.
- [128] R. L. Abduljabbar, H. Dia, and P.-W. Tsai, “Unidirectional and Bidirectional LSTM Models for Short-Term Traffic Prediction,” *Journal of Advanced Transportation*, vol. 2021, pp. 1–16, Mar. 2021.
- [129] Y. Du, Y. Fu, and L. Wang, “Skeleton based action recognition with convolutional neural network,” *Proceedings - 3rd IAPR Asian Conference on Pattern Recognition, ACPR 2015*, pp. 579–583, 2016.
- [130] Y. Hou, Z. Li, P. Wang, and W. Li, “Skeleton Optical Spectra-Based Action Recognition Using Convolutional Neural Networks,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 3, pp. 807–811, Mar. 2018.
- [131] J. Liu, A. Shahroudy, D. Xu, and G. Wang, “Spatio-Temporal LSTM with Trust Gates for 3D Human Action Recognition,” in *Lecture Notes in Computer Science (including subseries*

Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 9907 LNCS, 2016, pp. 816–833.

- [132] H.-H. Phan *et al.*, “Key frame and skeleton extraction for deep learning-based human action recognition,” in *2021 RIVF International Conference on Computing and Communication Technologies (RIVF)*, 2021, pp. 1–6.
- [133] K. Hu, F. Zheng, L. Weng, Y. Ding, and J. Jin, “Action recognition algorithm of spatio-temporal differential lstm based on feature enhancement,” *Applied Sciences (Switzerland)*, vol. 11, no. 17, 2021.
- [134] J.-Y. He, X. Wu, Z.-Q. Cheng, Z. Yuan, and Y.-G. Jiang, “DB-LSTM: Densely-connected Bi-directional LSTM for human action recognition,” *Neurocomputing*, vol. 444, pp. 319–331, Jul. 2021.

List of Publications

- 1) Bhogal, R.K., Devendran, V. (2022). Object Recognition Using Semantic Segmentation. In: Pandit, M., Gaur, M.K., Rana, P.S., Tiwari, A. (eds) Artificial Intelligence and Sustainable Computing. Algorithms for Intelligent Systems. Springer, Singapore. https://doi.org/10.1007/978-981-19-1653-3_38
- 2) Bhogal, R.K., Devendran, V. (2023). Action Segmentation for RGB Video Frames Using Skeleton 3D Data of NTURGB+D. In: Buyya, R., Hernandez, S.M., Kovvur, R.M.R., Sarma, T.H. (eds) Computational Intelligence and Data Analytics. Lecture Notes on Data Engineering and Communications Technologies, vol 142. Springer, Singapore. https://doi.org/10.1007/978-981-19-3391-2_15
- 3) Bhogal, R.K., Devendran, V. (2023). Human Activity Recognition Using LSTM with Feature Extraction Through CNN. In: Zhang, YD., Senjyu, T., So-In, C., Joshi, A. (eds) Smart Trends in Computing and Communications. Lecture Notes in Networks and Systems, vol 396. Springer, Singapore. https://doi.org/10.1007/978-981-16-9967-2_24
- 4) R. K. Bhogal and V. Devendran, "Motion Estimating Optical Flow for Action Recognition : (FARNEBACK, HORN SCHUNCK, LUCAS KANADE AND LUCAS-KANADE DERIVATIVE OF GAUSSIAN)," 2023 International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT), Bengaluru, India, 2023, pp. 675-682, doi: 10.1109/IDCIoT56793.2023.10053515
- 5) R. K. Bhogal and V. Devendran, "Action recognition for multiview skeleton 3d data using nturgb + d dataset," Computer Systems Science and Engineering, vol. 47, no.3, pp. 2759–2772, 2023

Chapter 37

Object Recognition Using Semantic Segmentation



Rosepreet Kaur Bhogal  and V. Devendran 

1 Introduction

Computer vision is a multi-disciplinary domain. Computer vision provide us with the structure, for automatic extraction, analysis, and comprehension from a single image or image sequence. Recognizing scenes from images or videos is one of the most real-world challenging application of computer vision in various spheres like industry, academia, security agencies, consumer agencies, and general-purpose, etc. Segmentation is one of the challenging tasks and it can be the first step in recognition of any task under preprocessing technique. The benefits of visual or group perception have been one of the most important computer vision problems used in various applications such as automatic driving [1] human machine communication, computer graphics [2], age-old search engines, and augmented to new reality.

The semantic segmentation can handle all deep learning architecture that has been used for classification. Finding objects from the image or videos can be the first step for classification. This not only provide us with the classes but also other information which may be useful for the recognition of objects with spatial information. Its goal is to label every pixel from the images or videos [3]. Per-pixel classification will be an effective and useful technique for various applications [1–3].

Segmentation is one of the most important techniques for image or video processing. Semantic segmentation served the front-end processing for computer vision applications. There is a different existing technique that relies on a database, a simple approach based on oriented pattern besides the use of color of texture [4]. Some methods, which are trying to train from end-to-end CNN networks, for multiple groups where co-segmentation can be possible to obtain the foreground of

R. K. Bhogal (✉)

School of Electronics and Electrical Engineering, Lovely Professional University, Phagwara, India
e-mail: rosepreetkaur12@gmail.com

V. Devendran

School of Computer Science and Engineering, Lovely Professional University, Phagwara, India

Action Segmentation for RGB Video Frames Using Skeleton 3D Data of NTURGB+D



Rosepreet Kaur Bhogal  and V. Devendran 

Abstract Action segmentation or video segmentation which used to extract action from video frames. It plays role in various applications, i.e., visual effect assistance in the movies, scene understanding in detail, virtual background creation, and a design CAD system that can identify automatically human action from videos without any object interference. This paper presents a system that automatically segments actions from videos. The window size is variable and depends on input video. The dataset used to show experimental data is NTURGB+D. The action segmentation has been shown using 3D skeleton information on RGB videos of NTURGB+D. The experimental results have shown the performance and it test results on 5 random action videos.

Keywords Segmentation · Actions · 3D skeleton data · RGB · Videos · NTURGB+D

1 Introduction

Identification of motion from video frames consists of various steps as preprocessing, segmentation, feature calculation, feature dimension reduction, and classification. The probability is high for a good recognition rate if feature calculation has been done for a properly segmented image/video frame. Either use any technique or deep learning approaches which are called edge technology nowadays. However, deep learning model designing has two main things. First, what input has been used to train or test a network and tuning of hyperparameter. If the input is not given appropriately, then the system may not recognize a few actions which are expected to do by the model. So, action segmentation can be a very important task for any designed model.

R. K. Bhogal (✉)

School of Electronics and Electrical Engineering, Lovely Professional University, Phagwara, India
e-mail: rosepreetkaur12@gmail.com

V. Devendran

School of Computer Science and Engineering, Lovely Professional University, Phagwara, India

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2023

203

R. Buyya et al. (eds.), *Computational Intelligence and Data Analytics*,
Lecture Notes on Data Engineering and Communications Technologies 142,
https://doi.org/10.1007/978-981-19-3391-2_15

Human Activity Recognition Using LSTM with Feature Extraction Through CNN



Rosepreet Kaur Bhogal  and V. Devendran 

Abstract Human activity recognition is important for detecting anomalies from videos. The analysis of auspicious activities using videos is increasingly important for security, surveillance, and personal archiving. This research paper has given a model which can recognize activities in random videos. The architecture has been designed by using BiLSTM layer which helps to learn a system based on time dependencies. To convert every frame into a featured vector, the pre-trained GoogLeNet network has been used. The evaluation has been done by using a public HMDB51 data set. The accuracy achieved by using the model is 93.04% for ten classes and 63.96% for 51 classes from same data set only. Then, this network is compared with other state-of-the-art method, and it proves to be a better approach for the recognition of activities.

Keywords Action recognition · HMDB51 · Neural network · CNN · LSTM · BiLSTM · Video frames

1 Introduction

Computer vision is a multi-stage domain, which essential framework for automatic extraction, analysis, and comprehension from a single image or image sequence. Man-made visualization from videos is one of the real-world challenges of computer vision in various sectors like industrial, educational, security, consumer etc. In the process of recognition, detection, or tracking, there are two broad terms “actions” and “activity”, which are generally used in the vision of the survey. And, there is difference

The original version of this chapter was revised: The incorrect last line in the Abstract has been removed. The correction to this chapter can be found at https://doi.org/10.1007/978-981-16-9967-2_76

R. K. Bhogal (✉)

School of Electronics and Electrical Engineering, Lovely Professional University, Phagwara, India
e-mail: rosepreetkaur12@gmail.com

V. Devendran

School of Computer Science and Engineering, Lovely Professional University, Phagwara, India

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2023, corrected publication 2023

Y.-D. Zhang et al. (eds.), *Smart Trends in Computing and Communications*, Lecture Notes in Networks and Systems 396, https://doi.org/10.1007/978-981-16-9967-2_24

Motion Estimating Optical Flow for Action Recognition

(FARNEBACK, HORN SCHUNCK, LUCAS KANADE AND LUCAS-KANADE DERIVATIVE OF GAUSSIAN)

Rosepreet Kaur Bhogal, V Devendran[#]
Lovely Professional University

Phagwara, India

rosepreetkaur12@gmail.com, svdevendran@gmail.com[#]

Abstract— Motion estimating is one of the methods which determines the movement from one frame to another in the videos. For an application of action recognition, choosing the optical flow can be an essential feature for recognizing actions. The optical flow consists of the information of the moving subject and objects in the video frames. This paper analyzes four motion estimating optical flow methods (Farneback, Horn Schunck, Lucas Kanade, and Lucas-Kanade Derivative of Gaussian) explored based on visualization and PSNR. The NTURGB+D dataset uses for the analysis of experimental results.

Keywords—Optical Flow, Action Recognition, Videos, NTURGB+D, Motion Estimation

I. INTRODUCTION

Human action recognition is one of the most challenging tasks nowadays. Various methods have been proposed in [1], which can use for pre-processing and recognizing activities. Motion estimation features are one of the main steps required to know activities in video sequences. The optical estimation technique gives in [2]. The optical flow gives motion information and helps to find an interesting point used to recognize actions. There are various non-parametric methods in which motion descriptor calculation helps estimate flow between one and the next frame. The flow descriptors aggregate the histograms on the temporal axis. The gradient-based methodology improves action recognition tasks [3]. There are various other handcrafted features like SIFT, HOG, GIST, and MHI [4]. These features have more focus on spatial information but not that much on temporal characteristics. The human motion analysis features the identification of human body shape, the relation of motion from one frame to the next, and one aspect of human activity recognition [5]. The optical flow based on warping explores which provided high-accuracy in [6]. This method can reduce the angle errors while computing optical flow. The optical flow can be used for video object segmentation, as mentioned in [7]. This model is attention based, which can use for object detection. Human activity recognition for video surveillance can design using an optical flow feature. Optical flow features include information related to the movement of subjects [8].

The mainly used optical flow estimation algorithm explore and analyse in the paper. Section 2 includes the research methodology. Section 3 includes experimental works, which show analysis based on a motion by considering random 15 RGB videos of NTURGB+D. Further, the conclusion of the

paper is which algorithm can prefer by the researcher for the estimation of activity.

II. METHODOLOGY

To calculate optical flow, techniques analyses in this paper. The methods are Farneback [9], Horn Schunck [10], Lucas Kanade [11], and Lucas-Kanade Derivative of Gaussian [11].

The HOF features are widely used for optical flow estimation by differential equations. Despite calculated differences, the measurement can include three stages of processing. First, pre-filtering to extract interest points. Second, compute the spatial-temporal derivative (called velocity vectors). Third, the integration of measurements to produce flow fields [12]. The methods used to find the object's direction and object's speed of moving from one frame to another.

For computation of the optical flow features can use the equation as follows:

$$I_x u + I_y v + I_t = 0 \quad (1)$$

In equation, I_x , I_y and I_t are the brightness derivatives.

A. Algorithm 1: Horn Schunck [11]

The method estimates a velocity vector by considering the flow features are smooth across the whole image.

$$E = \iint (I_x u + I_y v + I_t)^2 dx dy + \alpha \iint \left\{ \left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial u}{\partial y} \right)^2 + \left(\frac{\partial v}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 \right\} dx dy \quad (2)$$

In the above equation 1, the spatial derivative of optical velocity components is $\frac{\partial u}{\partial x}$ and $\frac{\partial u}{\partial y}$, α is the scaling factor that is related to smoothness. The method used to minimize equation 2 for each pixel in the frame with equations 3 & 4 is as follows:

$$u_{x,y}^{k+1} = u_{x,y}^{-k} - \frac{I_x [I_x u_{x,y}^{-k} + I_y v_{x,y}^{-k} + I_t]}{\alpha^2 + I_x^2 + I_y^2} \quad (3)$$

$$v_{x,y}^{k+1} = v_{x,y}^{-k} - \frac{I_y [I_x u_{x,y}^{-k} + I_y v_{x,y}^{-k} + I_t]}{\alpha^2 + I_x^2 + I_y^2} \quad (4)$$



ARTICLE

Action Recognition for Multiview Skeleton 3D Data Using NTURGB + D Dataset

Rosepreet Kaur Bhogal^{1,*} and V. Devendran²

¹School of Electronics and Electrical Engineering, Lovely Professional University, Phagwara, 144411, India

²School of Computer Science Engineering, Lovely Professional University, Phagwara, 144411, India

*Corresponding Author: Rosepreet Kaur Bhogal. Email: Rosepreet.kaur@lpu.co.in; Rosepreetkaur12@gmail.com

Received: 29 July 2022 Accepted: 21 December 2022 Published: 09 November 2023

ABSTRACT

Human activity recognition is a recent area of research for researchers. Activity recognition has many applications in smart homes to observe and track toddlers or oldsters for their safety, monitor indoor and outdoor activities, develop Tele immersion systems, or detect abnormal activity recognition. Three dimensions (3D) skeleton data is robust and somehow view-invariant. Due to this, it is one of the popular choices for human action recognition. This paper proposed using a transversal tree from 3D skeleton data to represent videos in a sequence. Further proposed two neural networks: convolutional neural network recurrent neural network₁ (CNN_RNN₁), used to find the optimal features and convolutional neural network recurrent neural network network₂ (CNN_RNN₂), used to classify actions. The deep neural network-based model proposed CNN_RNN₁ and CNN_RNN₂ that uses a convolutional neural network (CNN), Long short-term memory (LSTM) and Bidirectional Long short-term memory (BiLSTM) layered. The system efficiently achieves the desired accuracy over state-of-the-art models, i.e., 88.89%. The performance of the proposed model compared with the existing state-of-the-art models. The NTURGB + D dataset uses for analyzing experimental results. It is one of the large benchmark datasets for human activity recognition. Moreover, the comparison results show that the proposed model outperformed the state-of-the-art models.

KEYWORDS

Activity; recognition; multiview; LSTM; BiLSTM; NTURGB + D

1 Introduction

Action is when we do something, especially when dealing with anything like an object or human. The goal of any human activity recognition system is to recognize ongoing activities from ongoing videos automatically. Recognition of human activities enables real-time monitoring of public places like airports and stations can monitor patients, children and elderly persons [1]. Vision-based activity recognition systems highly impact various motivating application domains, like behavioral biometrics, Content-based video analysis, security and surveillance, interactive application and environment,

