Implementation of Deep Learning on Prosthetic Knee for Human Activity Recognition Using IMU Time Series Data

Author

MUHAMMAD HUZAIFA ASIF

00000275008

Supervised by

Prof. Dr. Mohsin Islam Tiwana

MASTERS IN MECHATRONICS ENGINEERING,

DEPERTMENT OF MECHATRONICS ENGINEERING,

COLLEGE OF ELECTRICAL & MECHANICALENGINEERING,

NATIONAL UNIVERSITY OF SCIENCE AND TECHNOLOGY,
ISLAMABAD, PAKISTAN.

AUGUST 2022

THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS/MPhil thesis written by Mr. Muhammad Huzaifa Asif, (Registration No. 0000275008), of College of Electrical and Mechanical Engineering CEME has been vetted by undersigned, found complete in all respects as per NUST Statues/Regulations, is within the similarity indices limit and is accepted as partial fulfillment for the award of MS/MPhil degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in the said thesis.

Signature: ______________________________

Name of Supervisor ________________________

Date: ________________________

Signature (HoD): ________________________

Date: ____________________________________

Signature (Dean/Principal): __________________

Date: ____________________________________

Acknowledgements

First and foremost, glory and gratitude to God, Allah Almighty, for showering His blessings on me during my research work, allowing me to successfully complete the research.

I would like to express my heartfelt gratitude to Dr. Mohsin Islam Tiwana, Ph.D., Associate. Professor and HOD Research at NUST College of EME, for allowing me to conduct research and providing invaluable support in the process. His dynamism, vision, honesty, and inspiration have all left an indelible impression on me. Working and studying under his supervision was a great honour and privilege. I am thankful for everything he has done for me. I owe my parents a debt of gratitude for their devotion, prayers, care, and sacrifices in educating and training me for the future. My mother and father (late) deserve special thanks for their prayers, understanding, and ongoing concern, inspiration, and assistance in helping me finish this research project.

This work is devoted to my adored parents and cherished siblings, who have always encouraged me to do amazing things.

Abstract

The study presented interface of Deep learning for classification of Human activity Recognition for activities like running, walk, stair climbing and stair climbing down. Custom made dataset is used based on WISDM standard. Accordingly, CNN-2D algorithm was designed for training and classification of HAR. Time series data is input for CNN-2D architecture after conversion of 1D time series data into 2D in pre-processing stage before fed into CNN network. Classification was performed by firstly training on PC, later on testing was done raspberry pi by wireless transmission of input data to raspberry pi from data recording hardware. After classification of HAR on real-time data prediction was given. The state-of-the-art algorithm performs excellent on custom dataset and provides accuracy of 97.58%.

Key Words: *Human Activity Recognition (HAR), Inertial Measurement Unit (IMU), Convolutional Neural Network (CNN), Raspberry pi, Time series*

Table of Contents

Contents

List of Figures

List of Tables

Acronyms

ADL	Activity of Daily Living
BDPL	Back Drivable Prosthetic Leg
BDM	Bayesian Decision Making
CNN	Convolutional Neural Network
CSV	Comma Separated Values
DTW	Dynamic Time Warping
HAR	Human Activity Recognition
HD	Hip Disarticulation Prosthesis
HP	Hemipelvectomy Prosthesis
IMU	Inertial Measurement Unit
K-NN	k-Nearest Neighbor Algorithm
LPK	Leeds Prosthetic Knee
LSM	Least Square Method
PCA	Principal Component Analysis
SVM	State Vector Machine

Chapter 1: Introduction

Human activity recognition is basically recognition of different type of human body movement and after recognizing it with different methods extract useful information which could be processed for different clinical, research purposes. The recognition of these human movements requires many different types of methodologies, algorithms, hardware schematics and training datasets. HAR has a very deep impact in human activity to a high-level knowledge via raw sensors or video-based approaches.

With the increasing trend of smart wearable portable devices, Human Activity recognition (HAR) is emerging active field of research and development. Different types of embedded systems-based sensors are used to monitor and gather human activity-based data to perform processing for different types of machine learning and deep learning algorithms [1]. Human Activity recognition is basically used to devise a system to monitor the people daily activity to make clinically decisions , body movement abnormality detection ,healthcare evaluations, fall detection, activity reminders and other healthcare evaluations[2][3].HAR Basically plays important role in rehabilitation as it helps checking health outcomes and all functional diagnosis . Human Activity Recognition (HAR) applications provides quality of assisted living system by integrating low level sensors with high end application with different algorithms of machine learning and deep learning. As health is always a priority, HAR is playing a vital role in assisting and monitoring different physiological activities e.g., cycling, running, climbing to make analysis and optimization ,In general there are two main types of categories or classes of Human activities . First class includes basic and simple human activities w.e.t motion and current poster e.g., walking, running, standing, climbing etc. Whereas second class includes bit complex human activities e.g., cooking, reading, watching etc. [4].

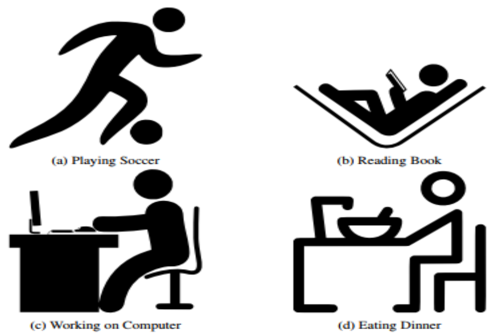

Figure 1-1 HAR Activities (a) [9]

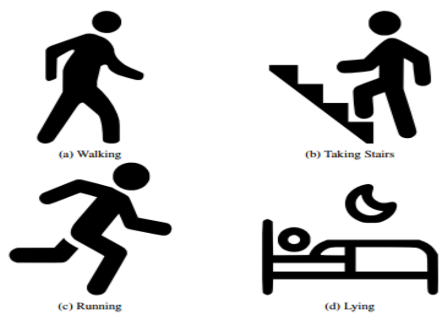

Figure 1-2 HAR Activities (b) [9]

Basically, there are two types of HAR methods used, Video based Human Activity Recognition and Sensor based Recognitions [5]. Video based HAR is based on images, videos by means of visual input while sensor based HAR is dependent of sensors or sensory networks e.g., accelerometer, gyroscope etc. Sensor based HAR is more reliable than video based HAR because of privacy protection, compatibility of data with less computational resources and time. With the increase in proliferation of sensor in smart devices our sensors can be integrated in different smart devices for data collection hence opening more gateways of research in field of HAR.

In the domain of research and development, Human Activity Recognition computation can be done with two types of approaches, Conventional pattern recognition and Deep learning. Conventional methods like SVM, hidden Markov's model etc. gave good results. These Conventional pattern recognition methods are dependent on hand crafted features which is basically a limited by human factor [6]. Whereas Deep learning methods are not limited by handcrafted features and can perform good in terms of classification and accuracy by learning useful features [7]. Hence Deep learning is best suited for classification for HAR [8].

1.1 Taxonomy of HAR:

The emerging field of human activity recognition is not new but has 3 decades of history. There is numerous research being carried out in this field from different perspective with different approaches and methodologies. Same as to have deep understanding of following subject, we would describe some important terminologies being used ahead in literature. We aim to discuss important work been done in this field and their approaches. Though in depth and in details following are discussed and established in respective areas [10][11].

Terminology:

1.1.1 Action:

Movement of person

1.1.2 Activity:

A state of action been performed by person

1.1.3 Physical Activity:

According to WHO, “Physical activity is any degree of movement which results in skeletal muscles that performs work via exerting energy”

1.1.4 Exercise:

A properly planned and structured physical work or activities called exercise e.g., running, weightlifting etc.

1.2 Taxonomy of Human activity:

Based on following existing studies [12][13] the taxonomy of human activity is classified widely and studies w.r.t human activity recognition. As per suggestion of Study [12] there are seven activities in HAR which includes daily activity, exercise etc. In the following figure from left to right we witness simple activity to complex activity.

1.3 Literature Survey:

Following thesis consist up of two parts firstly it will cover sensor-based approaches for HAR and available datasets in this regard. Then in second phase it will present survey w.r.t recognition of human activity by conventional and new state of the art deep learning methods. So, literature review would be divided respectively via above-described arrangement.

1.3.1 Sensor Modality:

Human activity recognition is not restricted to sensors but there are many categories to classify it but reason to choose sensor modality for our Human activity recognition are following. Generally, there are two methods for HAR: Video based, Sensor Based [14]. We would be using Sensor based which focuses motion data from smart sensors e.g., IMU [14]. Following points are described based on which we selected Sensor based rather than video based [15].

1) Widely used, portable, wearable
2) Low Cost
3) Low power Consumption
4) High-capacity Miniaturization
5) Privacy factor

Sensor modalities are classified into following main categories [16]

1) Body Worn Sensor
2) Object Sensor
3) Ambient Sensor
4) Other Modalities

1.3.1.1 Body Worn Sensor:

Human activity recognition includes one of most important modalities known as body worn sensor. These sensors are used by person such as accelerometer, gyroscope, magnetometer etc. The following parameters are changed w.r.t to human body movement. Thus, these types of sensors translate human body movement in different values which can be later used according to respective needs. One of most important things here is that access to these sensors is widely available in smart phone, smart bands, headsets, helmets etc. Deep learning is mostly connected with HAR [17]. Accelerometer is mostly used. Accelerometer is commonly used with gyroscope and magnetometer. Mostly sports and ADL is recognized via these sensors. Earlier statistical and frequency features are extracted from physical activities but after introduction of these body worn senor approaches original signal is fed into unit network directly.

a. Accelerometer:

Accelerometer work on principle of accelerations as a device which measures acceleration. Its measuring units is Meter per second squat or G-Force. Range of different high values in Hz is used for sampling frequency. In order to recognize the human activities accelerometer are worn or attached at body. Thus, we get timeseries data via accelerometer. Following are placement mostly people used as per found in literature review

1) Waist
2) Arm
3) Ankle
4) Wrist

b. Gyroscope:

It's a device that measures angular velocity and orientation The unit of angular velocity is degree per second. It sustains sampling frequency ranging from 10 to hundreds of Hz. Gyroscope comes with integrated accelerometer. As in a package it is mounted on human part. Gyroscope gives three times sequences as it has three axes.

c. Magnetometer:

Magnetometer is commonly used as wearable sensor for human activity recognition in a combination with accelerometer and gyroscope embedded into a combined small inertial unit. Its unit is tesla and sampling rate is same as described earlier. It contains three axes in a magnetometer.

d. Electromyography:

EMG sensors record and analyze electrical signal generated by muscles. EMG is different from above sensors because it is directly attached to human skin. So, it is not widely used with conventional scenarios , so used as fine surface e.g., hand and arm movement and facial expression.

e. Electrocardiography:

ECG is a biometric means of activity recognition that senses electrical signals generated via heart. It requires to be close contact with skin as different people heart work on different vibrations giving univariate time series data.

1.3.1.2 Ambient sensor:

These sensors play vital role in recording interaction between environment and human. There are many types of ambient sensors e.g., pressure sensor, sound sensor, radars, temperature sensors etc. Unlikely object sensors, these ambient sensors are used for capturing differential in environment. Mostly work done in regard of these ambient sensors are used to recognize daily human activity and gestures [20,21]. There is common trait among these sensors that they are difficult to deploy just like object sensors. These ambient sensors are very sensitive to environment thus are easily affected. There are few activities which are robust to environment

a. Wi-Fi:

It is a device based on local area wireless network. It uses transmitter to send signals and receiver to receive signals. The principle of using Wi-Fi in human activity recognition is w.r.t signal interference with human movement and location, thus based on received signal after direct propagation path and reflection, a signal is received. RSS of Wi-Fi is also used as standard for activity recognition [22]. There is an issue with RSS as is not stable even if there is not interference in environment. So as a replacement an advanced version of Wi-Fi signal metric Channel state information (CSI) a state-of-the-art approach is widely used now a days for activity recognition [23]. It’s up to such accuracy that lip movement [24], keystrokes [25], heartbeats can also be measured.

b. Radar:

As compared to Wi-Fi and RFID radar antennas are placed on same side of user as in WIFI and RFID both are opposite. Radar basically works on Doppler effect , as currently in research spectrograms are used to represent Doppler effect , after that fed into deep learning to process spectrogram.

1.3.1.3 Object Sensor:

Object sensors are used for specific object movement detection [26].As compared to body worn sensors these are used fir detection if movement of object w.r.t human activities. For example, if it is attached to glass of water , if its accelerometer it will give drinking activity. Most common examples are RFID. RFIDs are capable of giving fine processed information for any complex activity. These object sensors are not likely to be used with other sensors e.g., body worn sensors because of complexity in deployment. With that as an issue , it provides opportunity to more researcher to work on combination of these sensors for high level activity performance.

a. RFID Sensor:

It basically uses electromagnetic field for identification if tags attached to objects automatically. As in this tag, information is stored. There is main two types of RFIDs Active & Passive. Among these active RFIDs are dependent on local power source to keep emitting their signals which can be read by RFID even on long distance. As compared to active, passive RFIDs collect energies from nearby available RFIDs to work. As per their cost, passive is cheap as compared to active and light. RSS is also deployed w.r.t RFID for HAR [27].

1.3.1.4 Other Modalities:

Other than the above modalities discussed in detail following are some important modalities used in literature by different researchers.

1. Audio sensor:

Audio sensors can be used for human activity recognition as its also available normally in smartphones. As we know that basic principle of the speaker is to throw Ultrasound signals while the microphone receives these signals. So, point to be considered here is ultrasound would be modified once there is human activity. Fine-grained types of movement are detected. One of the most important works done was that ultrasound signal was used from pair of microphones and speaker to categorize or recognize chewing activity [28]

2. Pressure Sensor:

Pressure sensors require mechanical mechanisms such as direct physical contact with the human body for activity recognition. It can be installed in any wearable or smart environment. In case of deployment in the environment, chairs, beds, floor [29,30] etc. are examples of entities for activity recognition. Due to direct contact with human body, minimal movement can be observed, hence we can say that its suitable for exercising etc. monitoring . Similarly, in case of wearable device pressure sensors can be used to harvest energy thus making self-powered standalone system. Their location of installation is shoes, human chest and wrist band etc. [31].

1.3.2 Deep learning for HAR:

In general, deep learning is branch of machine learning that possess different multiple layers which process & extract large scale data. As per following studies, [25][32] Deep learning methods were able to achieve better accuracy and robustness, performance in contrast with traditional methodologies. The basic common traits among traditional approaches, common issue was shallow features problem. As per sensor modality, HAR can be done by processing information from inertial sensors e.g., accelerometer, gyroscope, etc. As now a days these types of sensors are common in smart phone so our smart phones can be used to collect.

1.3.3 Convolutional Neural network:

CNN constitutes important parameters e.g., Equivariant representation Parameter sharing, sparse interaction. In the structure of CNN, after the data passed through convolution the very next step is pooling and full connected layers which classify different activities. In terms of extraction of features from signals, CNN has given very good results e.g., classification, text analysis, speech recognition etc. The important topic like HAR which is covered in following thesis when processed w.r.t to CNN, it gives two advantages w.r.t other models e.g., scale invariance and local dependency. By Local dependency we usually mean that nearby signals present for HAR would be correlated while it is scale invariant w.r.t different frequencies.

Following aspects are considered when selecting CNN for HAR

a) Input adaption
b) Pooling
c) Weight sharing

a) Input Adaption

Basically, for Human activity recognition, mostly sensors give reading in time series e.g., acceleration signal. As it is multidimensional 1D temporal reading so we can say that input adaption is important as main purpose to have input adaption is to have deep insights if virtual image been created by it. Model driven, data driven are two main types for it.

- **Data Driven**

The basic idea of data driven is such that each dimension is treated as a channel before performing the convolutional as after convolution, pooling each of these channels are flattened. For example, a work done, as a one channel each of dimensions was treated as for accelerometer for example RGB image, after convolution, pooling was applied but separately. Similarly resized kernel was made to obtain best results if CNN via 1D convolution for HAR data. In short data driven approach consider each sensor 1 D reading as an image as it is easy to process by this way. Only issue remains, left out of dependencies sensors and dimensions which cause problem in performance sometime.

- **Model-Driven**

Basically, this type of adaption converts inputs to 2D image so that 2D Covnets could be applied. It falls under complexity as it's a non-trivial combined with its dimensions to from an image as it's a complex work to transform times series data into an image. Similarly, via modality transformation pressure sensor data was transformed to image. As for this type of driven method , domain knowledge is needed as its non-trivial task.

b) Pooling

Pooling is used as a combination with CNN in common practice as for this max pooling and average pooling done after convolution. The basic purpose of pooling is to avoid overfitting, and speed -up training on large data.

c) Weight Sharing

Weight sharing is done to complete training in less time as it uses relaxed partial weight sharing approach. As per literature, partial weight sharing is very beneficial as it could improve performance of CNN.

1.3.4 Applications:

HAR is not the limit of research-based purpose but to implement it for daily usage of people and contribute towards society. As most of work in this field is focused on Activity of Daily living and daily life sports [33]As its obvious that if movement s is simple, it would be easier to capture, especially when body worn sensors are used. For some more deep research purpose in some sides sleep, respiration activity recognition is performed to detect complex movements e.g., usage of object with ambient sensors e.g., sound sensors and Wi-Fi. Similarly, as per increase in trend in HAR for health care purpose, in some studies Parkinson disease, trauma, PAF etc. were also dealt with situation. With the strong argument based on research, disease problems are very much respectively related to certain type of different body movement. Hence sensors are used. To have a deep research in Health care via HAR, it's important to understand that there is a correlation between disease and body movements. As if its more inconsideration it would be beneficial, the most important thigs here is the usage if proper sensors for proper purpose at proper location. For example, for Parkinson disease, in literature, inertial sensors are used in shoes as w.r.t frozen of gait[20].

Similarly, in regard of other health care conditions it's important to know its high-level information in terms of high-level activities as it would be very beneficial and resourceful for HAR. Similarly, parameter like environment, behaviors, movements are important part for HAR. Though there are also many other parameters playing their role for HAR, we hope in future all parameters' results would be translated into useful research thus increasing activity recognition accuracy by considering multiple parameters.

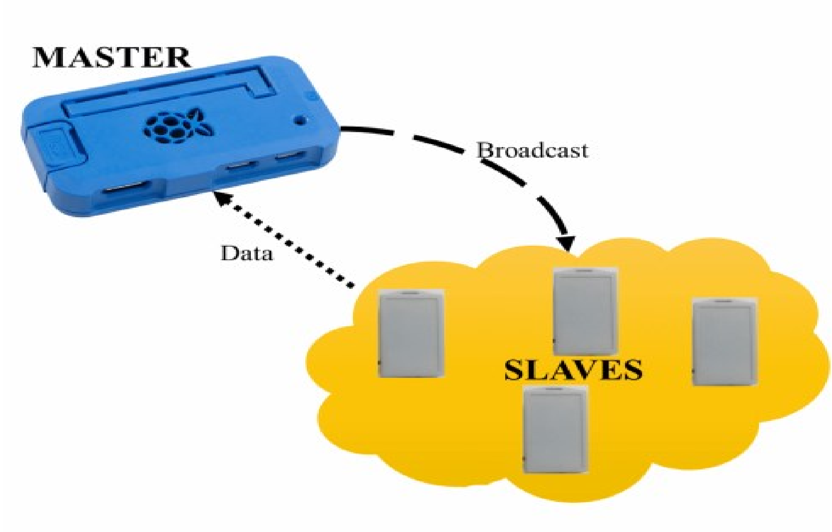

Figure 1-3 Master-Slave Combination of Sensor [34]

Similarly, In [34], a challenge was addressed about monitoring outdoor activities and

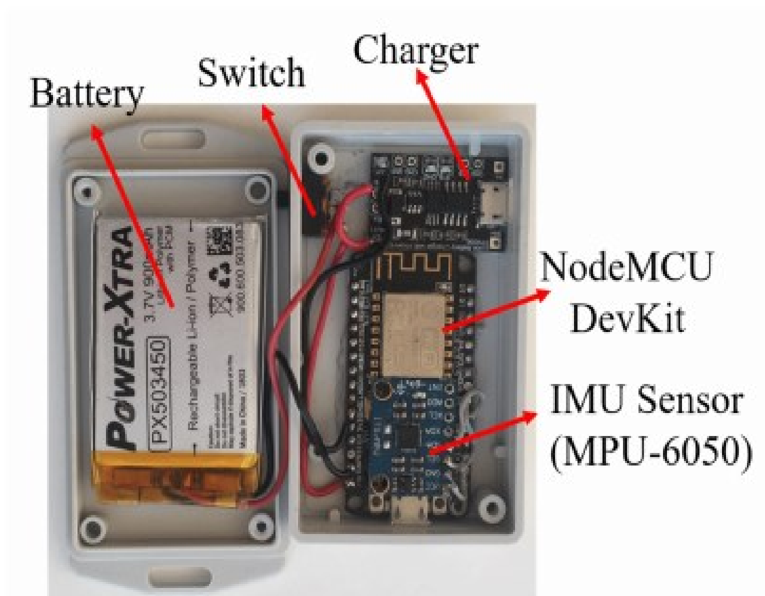

Figure 1-4 MCU & IMU Sensor[34]

extract the feature, so a challenge was addressed by creation of new dataset. A brief explanation was given by HAR needs to be shifted from traditional ML algorithms to Deep learning algorithms. Furthermore, combination if CNN LSTM was also explained for this purpose and its results. In the following fig 1-5[34] basic block diagram is shown

In fig1-5, fig 1-6, CNN , LSTM & CNN-LSTM architecture was explained in depth with input layers , feature extraction layers and classification layers.

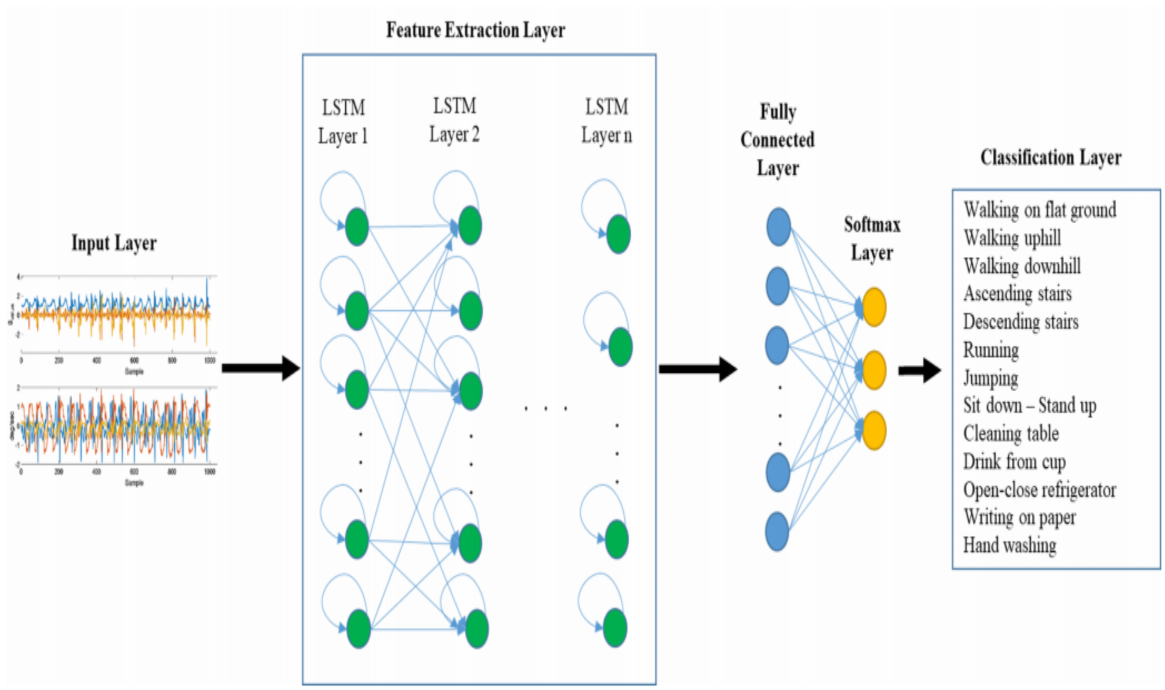

Figure 1-5 Layers of LSTM [34]

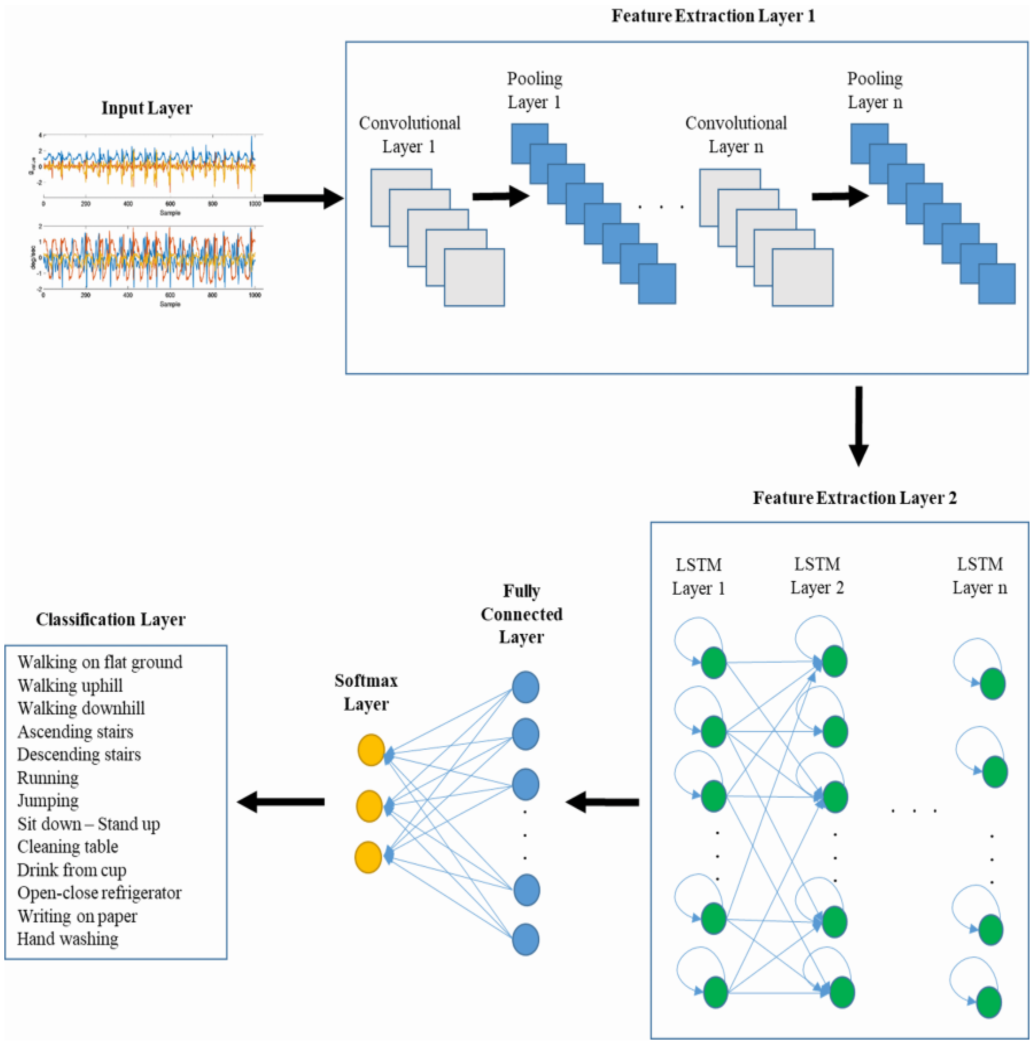

Figure 1-6 Demonstration of CNN-LSTM [34]

Similarly, in [35], CNN architecture was used for HAR on self-created dataset. The efficiency achieved was 93.8%. It Basically explained its better than deep neural network. Basic structure explained below in Fig1-7[35]

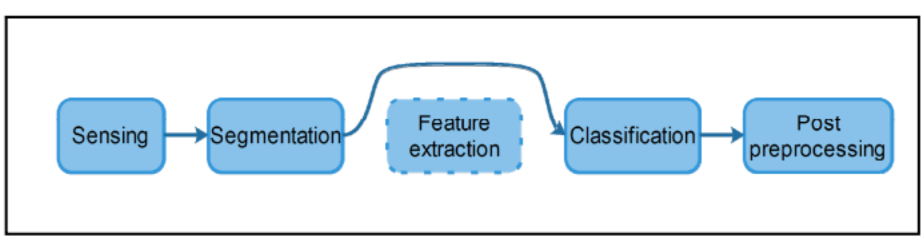

Figure 1-7 Classification Overview [35]

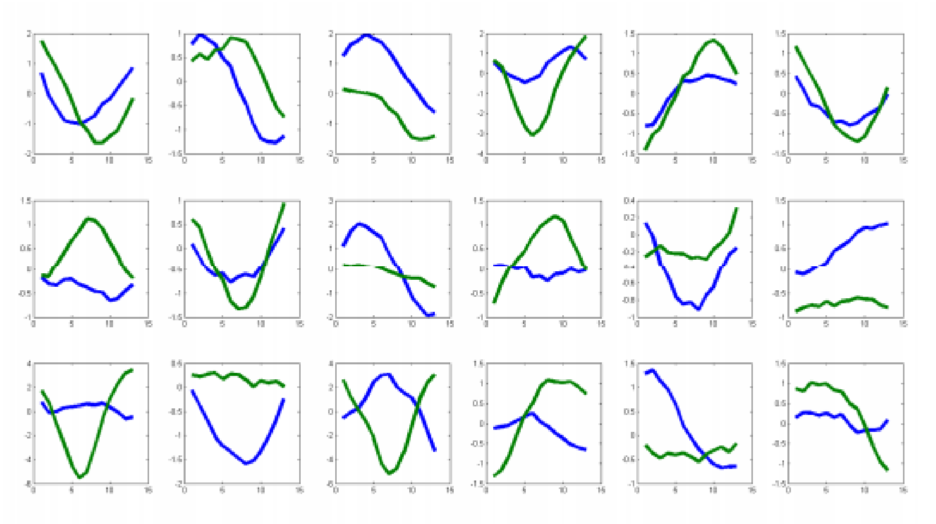

Figure 1-8 Classification Overview [35]

Similarly, in Fig 1-8[35], It may be seen that the primary convolution layer has found out varied relationships between axes, some area unit within the same phase, some with delay or maybe opposite in part. The acceleration signals obtained from totally different styles of activities are all composed of those basic options. In literature [36], a basic point of conflict is explained about handcrafted based HAR and Automated HAR. For better explanation, both differences are explained visually. As we can see that in Fig 1-9, the steps like highlighting segmentation, feature extraction while in Fig 1-10 steps used in Deep learning approach are explained in depth where features are product of hidden layers of NNs training phase and construction phase.

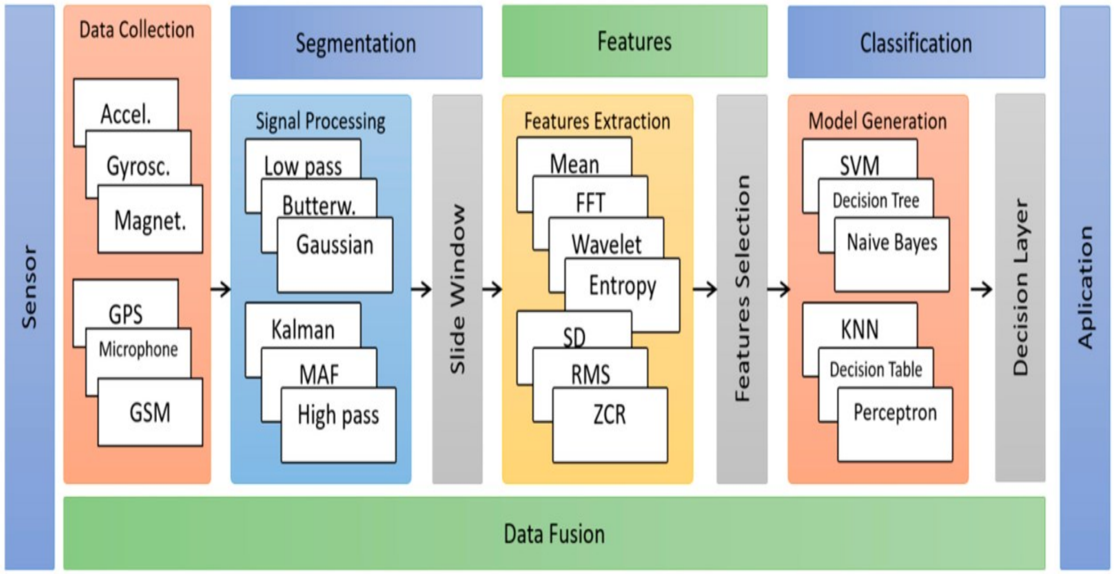

Figure 1-9 Manual Feature [36]

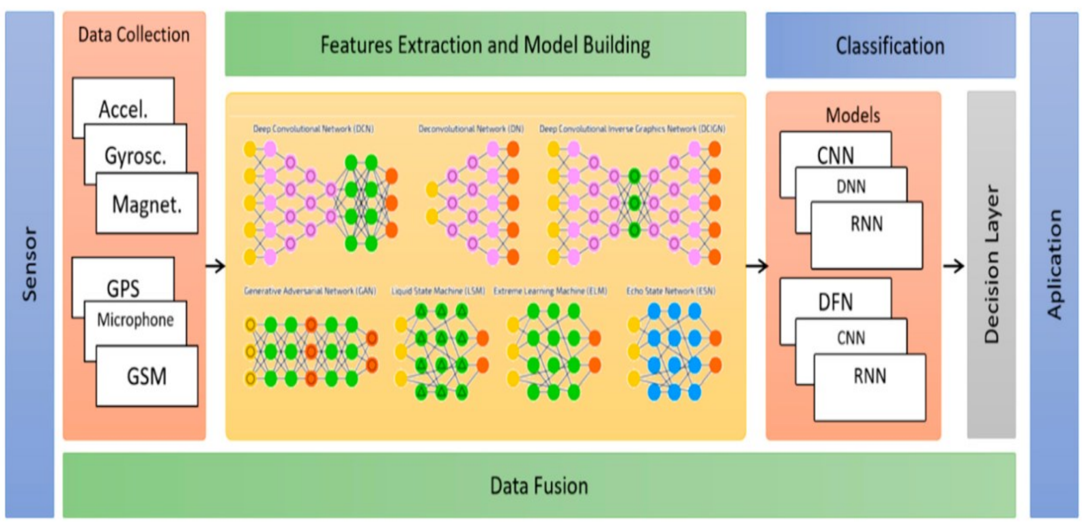

Figure 1-10 Automatic Features [36]

In [38],a method is proposed in which deep Convolutional neural network is used which would be automating feature learning from different raw inputs in very organized way. Through this architecture, the features which are learned are considered as High-level representation of low-level raw signal of time series. So, by using supervised leaning the features which are learned are more useful than before. By this overall feature learning with classification both are enhanced. Thus, it will outperform HAR other algorithms. In

Fig1-11, architecture used in following study is visualized whereas its performance measuring metrics.

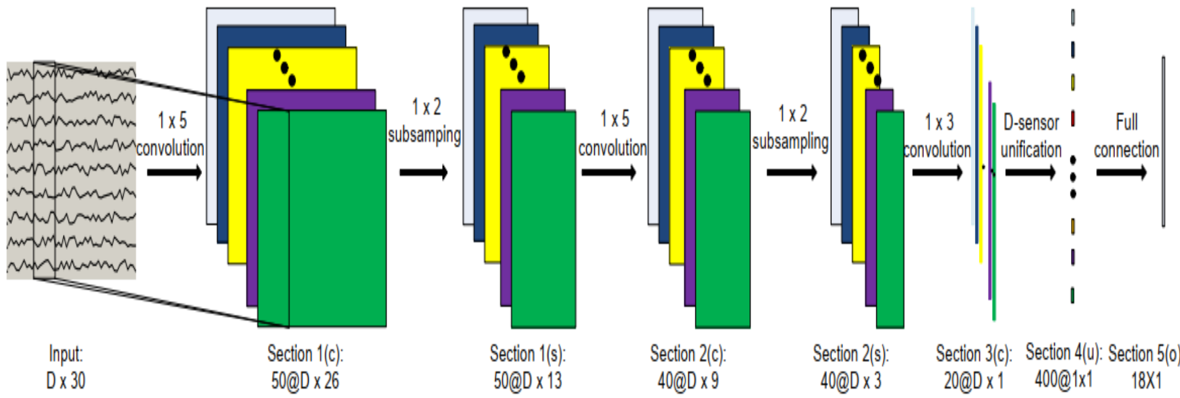

Figure 1-11 CNN Architecture [38]

Meanwhile the flow of paper discussion in continued, here one important thing about classification needs to be addressed as it is final phase in HAR. At this point a classifier which was trained, would now be used for classifying different type of activities. As in the modern age of computing where we have different means of computation, classification can be done via online and offline. Before jumping into classification computation let's have a look on its training phase for deeper understanding. As by online training its meant that classifiers would be on training on hosting device e.g., Raspberry Pi, cloud etc. While by offline we mean a desktop or machine used for training having all onboard components for processing. Raw data is not saved in online training for later on use but instead of that its immediately processed. In the following Fig 1-12 , a flow is shown

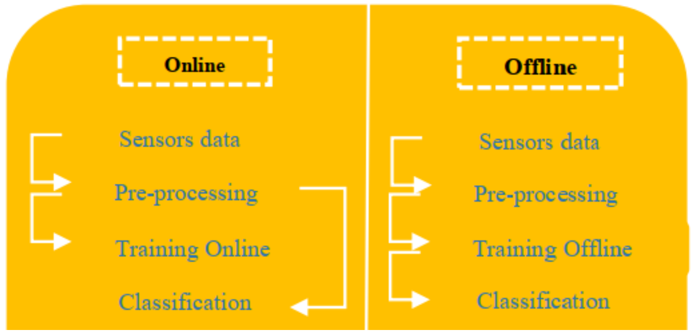

Figure 1-12 Online vs Offline [39]

As per applications, in following research article [39], online HAR method is very useful in healthcare as continuous monitoring is required w.r.t patients pathologies. Similarly, when Online HAR is not required based on criticality, Offline HAR is used. As in offline mode, user do not need to wait for response of feedback for recognition system, the recipient of results receives results after offline analysis and classification.

Recent years there was a large turnover from traditional to modern methods for new problems but in field of HAR it’s still in transition. As with the intervention of technology in our daily life like smart phones, these device has now been used for HAR as a primary source. Similarly, the basic goal of Har is usage of Machine learning models with good accuracy for HAR prediction. Deep learning a sub-branch of ML is highly used in HAR. After the analysis of different approaches in following article [39] , there are three important model flows:

a. **Traditional Model Flow:**
Data is collected via sensor ; features are extracted via traditional methods after that activity labels are extracted.

b. **Deep Learning Model Flow**
Data is collected via sensors and features are extracted but deep learning is used then after that finally extracted activity labels is generated.

c. **Deep Feature Extraction and Model building flow:**
Data is collected then deep learning technique is deployed for automatic feature extraction. Before final results SoftMax layer is used. In the following figure 1-13, both 3 methods are explained.

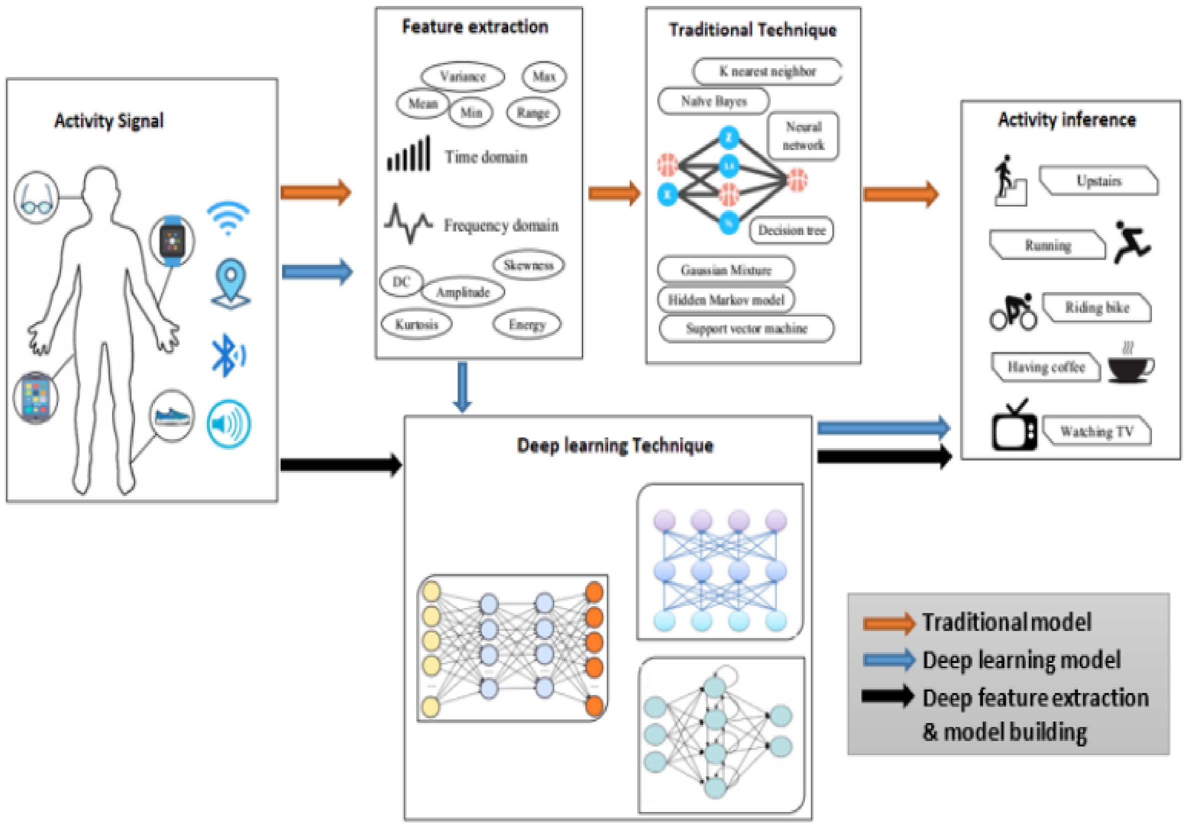

Figure 1-13 Models Flow & Summary [39]

Similarly, a traditional vs deep learning algorithms compared in graphical in following figure Fig 1-14[39]

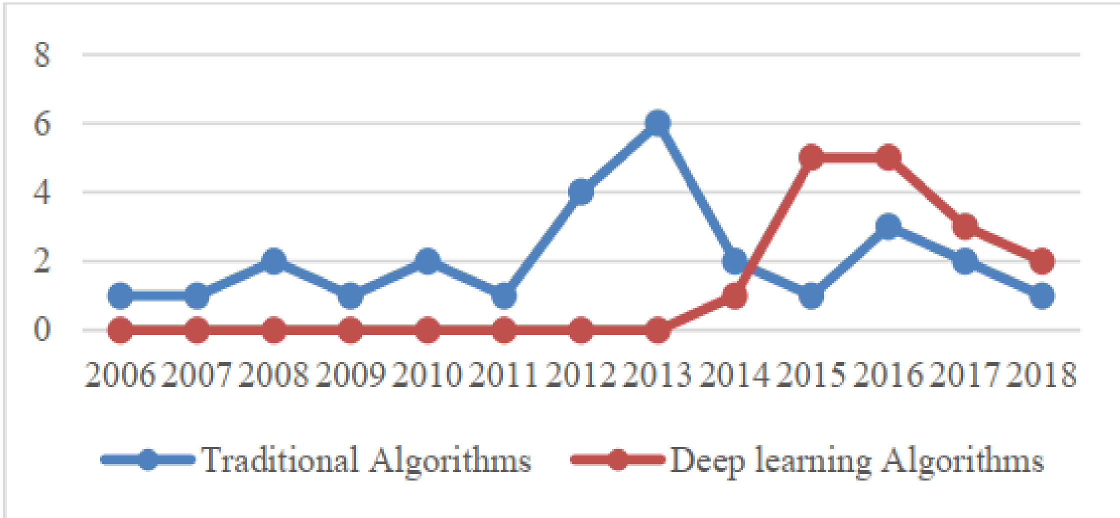

Figure 1-14 Traditional Vs Deep learning [39]

In the study [40], a user independent approach of deep learning for HAR is proposed. This was achieved by CNN. Convolution Neural Network played role for Local feature extractions, with statical features which help to preserve global time series information saving. Similarly, time series length was checked for recognition accuracy so that it could be limit up to one second only. Two most important data sets WISDM and USCI were used in this study. In following fig 1-15, we see basic CNN architecture used in this study.

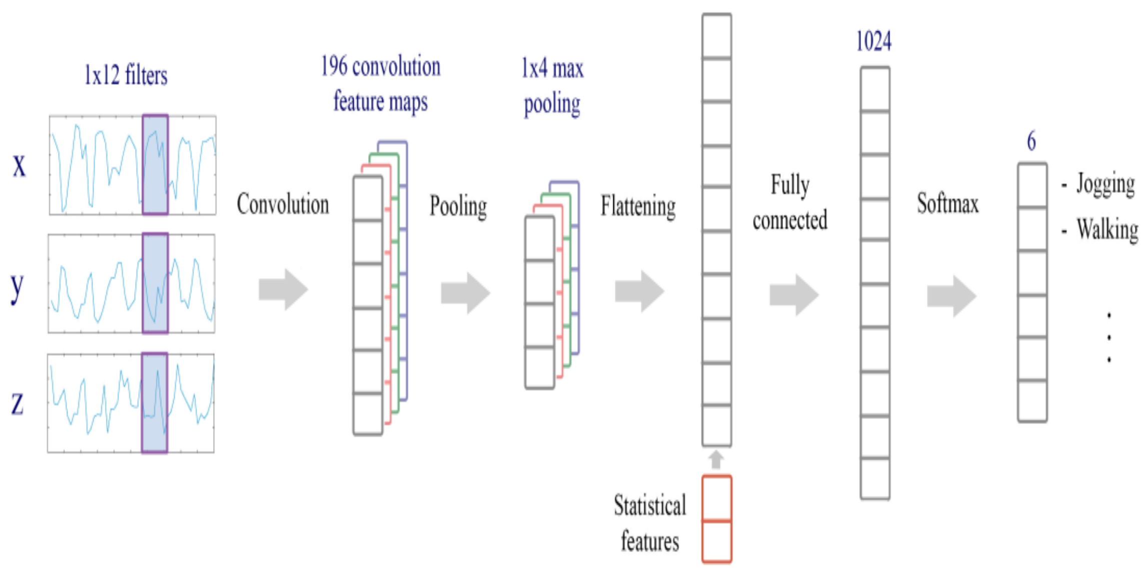

Figure 1-15 CNN architecture with Statistical Features [40]

Similarly, in the second part of following study, time series length was checked for recognition accuracy so that it could be limit up to one second only. Following Fig 1-16 shown below

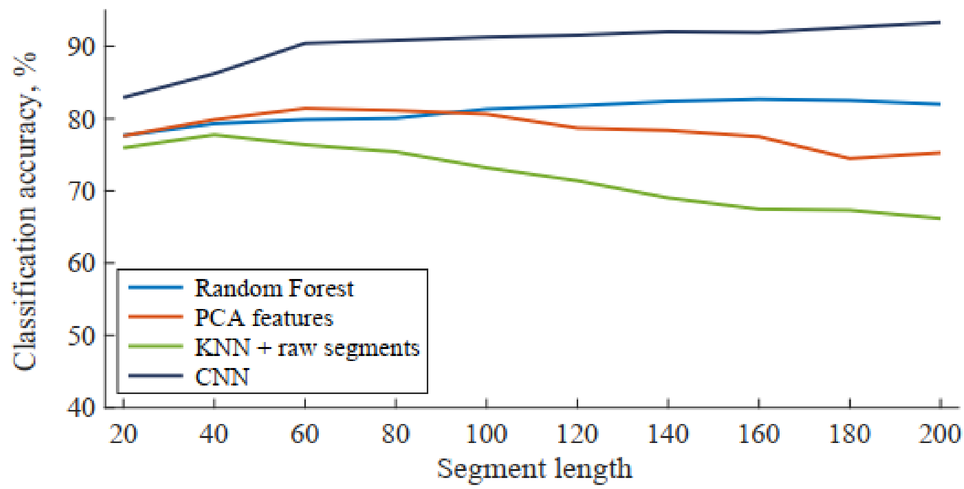

Figure 1-16 Classification Accuracy [40]

In [40], Human activity recognition CNN architecture is proposed which could handle different sequences of measurements from different body sensors each separately. This architecture was evaluated on scale via using standard datasets e.g., Opportunity, industrial dataset, Pamap2. As this architecture gave good results on these datasets. Similarly, different network combinations are evaluated. In contrast to following study Fig 1-17 shows an example is illustrated via IMU sensor measurement with CNN sliding window approach.

The CNN-IMU combination of architecture can be visualized in Fig 1-18, as m parallel branches, with one IMU each. Branches are B Blocks with [5x2] temporal convolution, [2x1] layer of max pooling. The output of these block is connected and forwarded as fully connected layer. Here output layer is SoftMax.

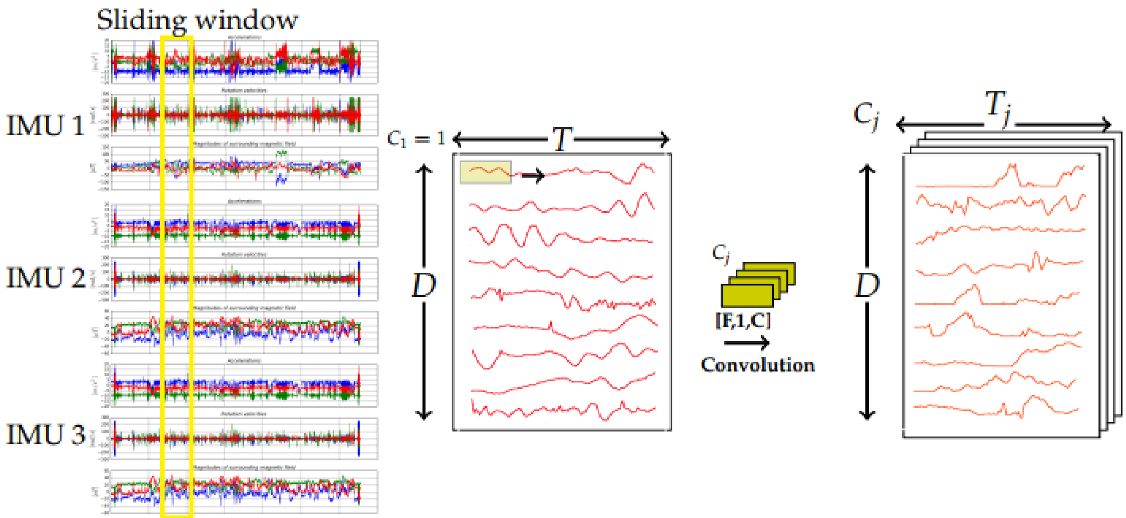

Figure 1-17 IMU Sensor & Sliding Window [40]

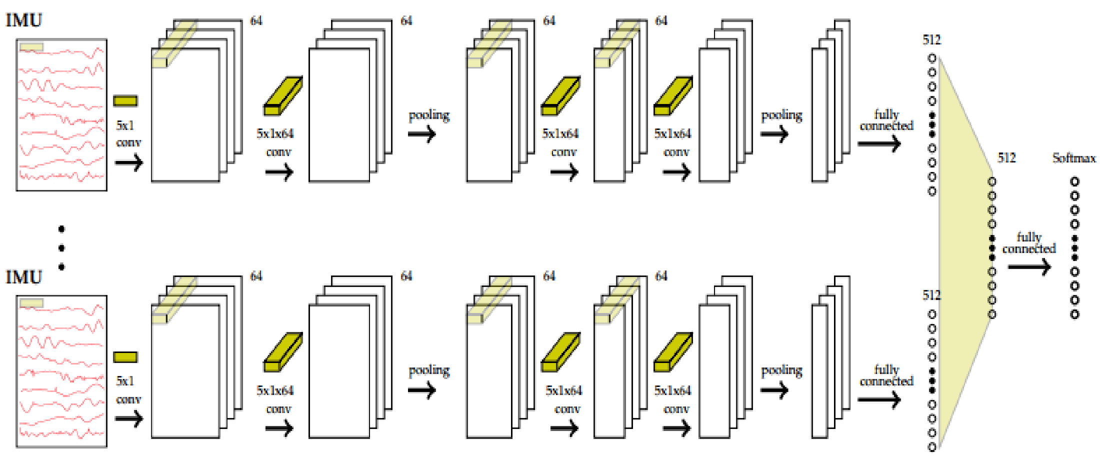

Figure 1-18 CNN-IMU Architecture [40]

In the following research article , the architecture that is proposed consist up of multiple time series w.r.t multiple IMU sensors. As in this system multiple sensors are attached to individual at multiple locations. Its main stand out work is the use of temporal convolution layers. By this temporal convolution, it targets temporal local features via input. Thus, the fully connected layers played their role and connect all local features to form global representation. This architecture consists up of multiple branches processing parallel to be on the side of wider network then deeper. Each branch represents single IMU sensor. Thus, IMU sensors are located on different parts of body as branches only process signal coming from respective body part thus increases descriptiveness.

As the proposed architecture in the following paper is mainly focused on standard dataset like Opportunity , Pamp2 , order picking data set. Additionally, an effect of usage of max pooling was also investigated. If the sequencies are long max pooling plays important tole. Thus CNN-IMU outperform other baseline CNN architectures by giving accuracy of 92.22%.

1.3.5 Benchmark Paper:

In [37], HAR was performed via smart phone data. In the following article, multivariate data was used to detect the activities. As we know that time series data has Local dependency by inherent. So, CNN was used to exploit these issues of characteristic and translate it according to time series. Thus, is following article its shown that by increasing the number of layers in CNN, CNN performance increases but the complexity of features which are derived decreases by adding each layer. Similarly, it was explained that blindly none of the parameters needed to be increased of decreases as preservation of information and transfer of information from one layer to other layers are also very important. It was also described that with increase if filter size, low pool size results can be improved. In the following Fig 1-20, overall CNN architecture was explained in depth. Whereas the data recorded shows in Fig 1-19.

Figure 1-19 Accelerometer Gyroscope Signal [37]

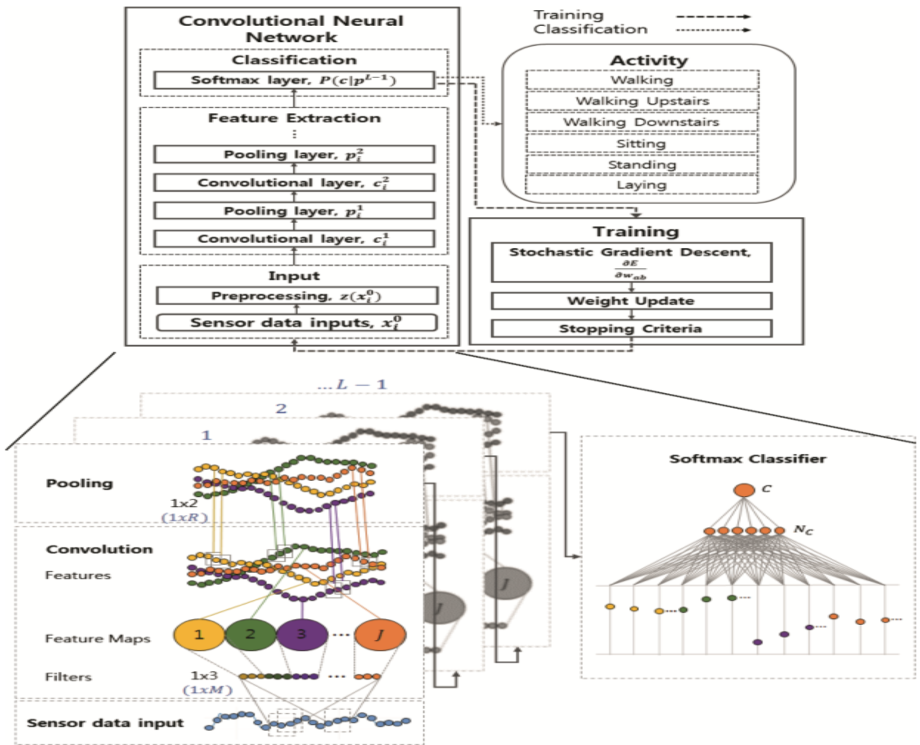

Figure 1-20 CNN Architecture and Training [37]

In terms of hardware and software, network size is basically limited by the amount of memory available in GPUs. Hypermeters were tunned via greedy-Weise tuning method in which feature maps, number, size of filter, pooling size. Overall, 94.70% of accuracy was achieved via following algorithm of CNN. Lastly, we can witness confusion matrix of following proposed CNN architecture in fig given below

		Predicted class						
		W	WU	WD	Si	St	L	Recall
Actual class	Walking	491	3	2	0	0	0	98.99%
	W. upstairs	0	471	0	0	0	0	100.00%
	W. downstairs	0	0	420	0	0	0	100.00%
	Sitting	0	0	0	436	34	21	88.80%
	Standing	0	1	0	24	496	11	93.23%
	Laying	0	0	0	43	23	471	87.71%
	Precision	100.00%	99.16%	99.53%	86.68%	89.69%	93.64%	**94.79%**

Figure 1-21 Confusion Matrix CNN [37]

Till then in the respective thesis, a detailed explanation was given with comparison to basic literature review. Now few researches would be discussed in detail. In [33],a 1D CNN framework was proposed for human activity recognition using tri-accelerometer . Data was collected using smartphones. In the approach of 1D CNN vector magnitude data of accelerometer data was used. So, to proceed for experiment, three kinds of human activity was recorded walking, running, staying still.

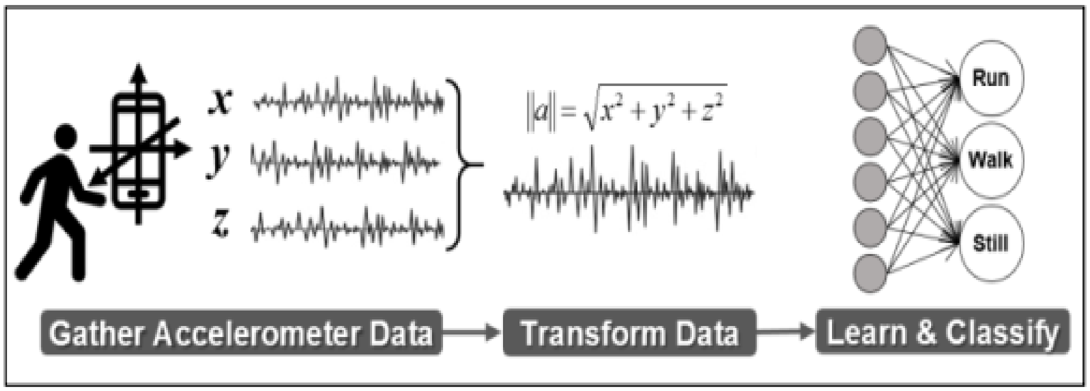

Figure 1-22 Workflow Demonstration [33]

From the following Fig 1-22 [33], a demonstration of work was shown that data was gathered from accelerometer and after transformation it sent to learning and classification. Similarly, in-terms of learning architecture a brief explanation is given in following Fig1-5[33]. The learning architecture used in this paper was CNN which would be classifying between 3 states e.g., running, walking, still. In the following CNN architecture, window sizes were 3,4,5. The size of stride for all windows was selected 1. 128 total filleters were utilized for vector creation. Following other max-pooling, dropout, output detail figure explained below.

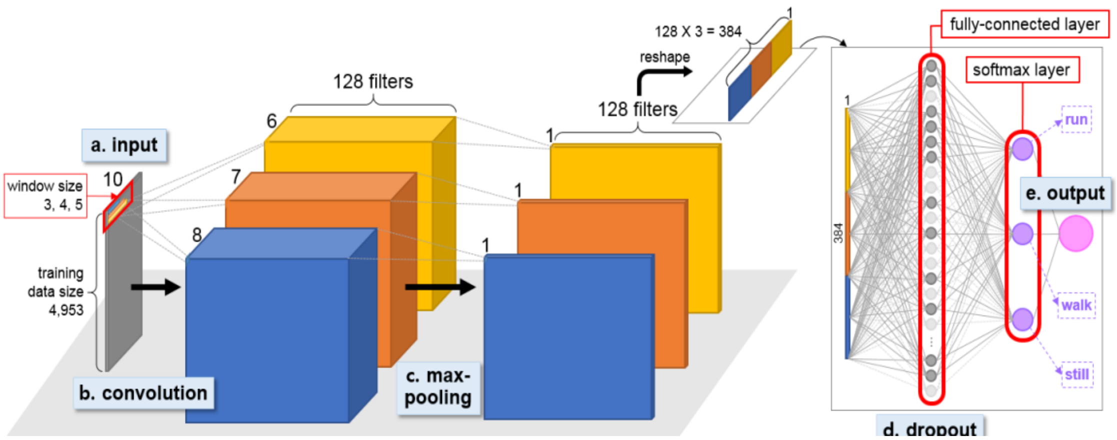

Figure 1-23 CNN Architecture [33]

Similarly, this article was mainly comparing with baseline random forest method with its own proposed 1D CNN. As for 1D CNN input vectors were fed via in two types of input vectors one with ten second and other with twenty second. Thus, later on both results of F10 and F20 were compared in result section with its parameter like Convolution window size as 3,4,5. By default stride as 1 with 128 convolution layers with dropout if 0.5. Training batch size was of 64 with 200 epochs. Loss function was computed via sigmoid cross entropy via ADAM as optimizer. Thus, a comparison is given below of confusion matrix.

Random Forest		**Predicted Class**			**Recall**
	Activity	***Run***	***Walk***	***Still***	
Actual Class	***Run***	**587**	100	4	84.95%
	Walk	65	**561**	65	81.19%
	Still	0	62	**629**	91.03%
Precision		90.03%	77.59%	90.11%	**85.72%**
1D CNN		**Predicted Class**			**Recall**
	Activity	***Run***	***Walk***	***Still***	
Actual Class	***Run***	**605**	84	2	87.55%
	Walk	68	**597**	26	86.40%
	Still	0	0	**691**	100%
Precision		89.90%	87.67%	96.11%	**91.32%**

Figure 1-24 F10 Confusion Matrix [33]

Random Forest		Predicted Class			Recall
	Activity	*Run*	*Walk*	*Still*	
Actual Class	*Run*	**522**	57	2	89.85%
	Walk	33	**530**	18	91.22%
	Still	0	80	**501**	86.23%
Precision		94.05%	79.46%	96.16%	**89.10%**
1D CNN		**Predicted Class**			**Recall**
	Activity	*Run*	*Walk*	*Still*	
Actual Class	*Run*	**527**	54	0	90.71%
	Walk	18	**560**	3	96.39%
	Still	0	52	**529**	91.05%
Precision		96.70%	84.08%	99.44%	**92.71%**

Figure 1-25 F20 Confusion Matrix [33]

Thus, confusion matrix in comparison to random forest was presented in this work. As it approached accuracy of 91.32% and 92.71% for F10, F20 respectively. In the following study we observe that there is problem in Walk activity classification as also described in results section of that study.

Similarly, in following study [41], a method based on CNN was developed which was able to deal with scale invariance and local dependency of signal. The experiment was performed on three public data sets like Skoda, Opportunity, Anti tracker dataset and gave good accuracy w.r.t existing methods. Mainly there would three main tasks of this study, firstly, an approach is proposed based on CNN for feature extraction without any prior knowledge. Secondly, the proposed methods would be dealing with scale invariance and local dependencies. Thirdly, experimental results are given on these 3 public datasets. Following figure shows detailed structure of human activity recognition.

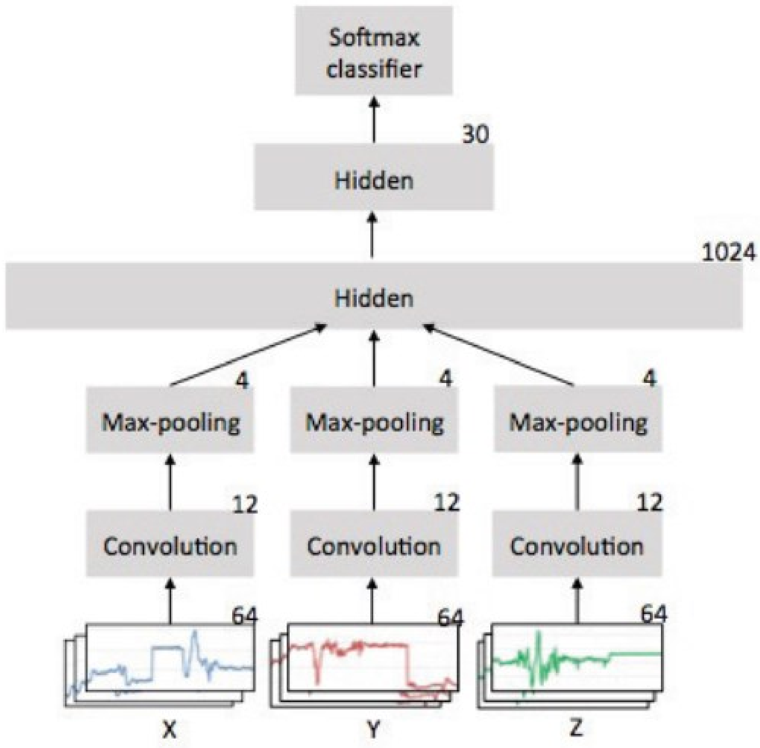

Figure 1-26 CNN structure proposed [41]

Similarly, from the results section we observed that the following CNN based architecture which performed on Anti-tracker has some errors in classification as indicated in its confusion matrix.

		Predict Class					
		Jog	Walk	Up	Down	Sit	Stand
Class Actual	Jog	667	5	1	3	0	0
	Walk	1	1145	8	5	0	0
	Up	5	13	274	17	1	1
	Down	2	9	13	231	0	0
	Sit	0	0	0	0	166	0
	Stand	0	0	0	0	0	133

Figure 1-27 Confusion matric on Anti-tracker [41]

If we analyze the following study in depth, we come to know that main problem of CNN architecture to classify is coming in walking activity especially walking up and walking down as per author reason it’s because these activities resemble each other because of which such problem was faced.

Chapter 2: Methodology

Human activity Recognition is a very broad term and has many appealing applications. As per thesis, Deep learning algorithm would be implemented hardware via Human activity recognition approach using IMU sensor.

In the following thesis to achieve our task we will start from grass root level and provide justification of selection any particular methodology to follow and its future prospects with respect to benefit of field.

2.1 Why Deep learning

In terms of selecting any Machine learning approach for our task, it was confirmed that we need to work on that methodology which could be efficient and give good result. As for this we had two option conventional pattern recognition methods and modern deep learning methods. No doubt that Conventional PR methods worked tremendously [41]. Previously the method to extract features were very difficult as it requires features to be deal in hand crafted and heuristic. Thus, it totally depends in that human experience and knowledge that how he drafts or handcraft the features. As this might be very useful is some specific task addition or plugin but on a broader spectrum, it's difficult as for general task it would have a low success rate for building activity recognition systems. A glimpse of conventional PR is shown in Fig 2-1.

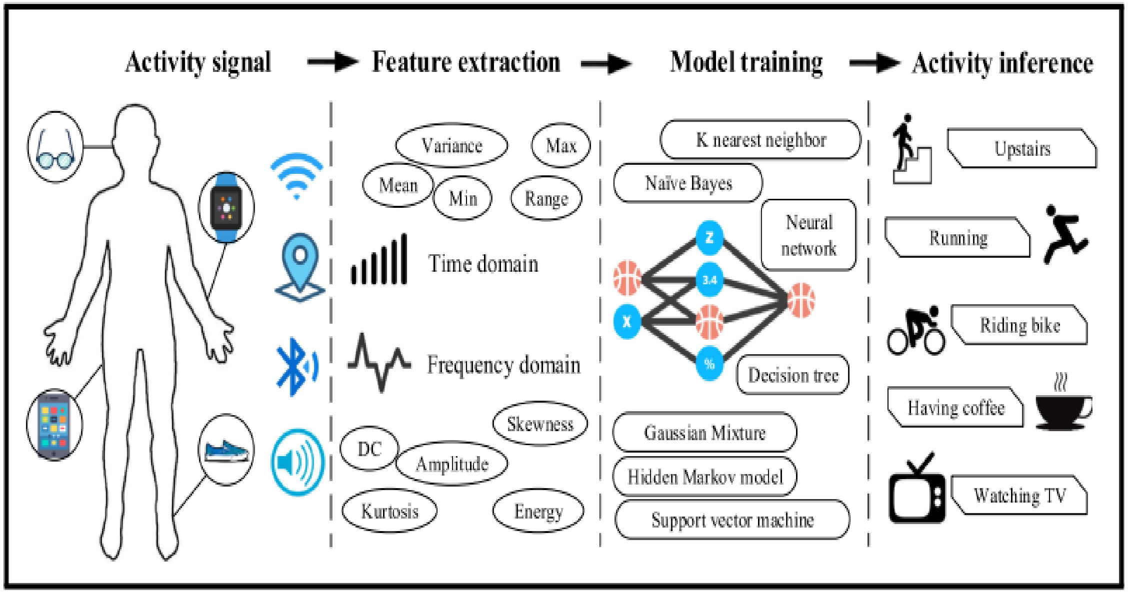

Figure 2-1 Conventional PR methods [42]

Similarly, as per human expertise and experience, only shallow features were able to be captured [66].By these shallow features we mean statistical data , e.g., variance , frequency , mean. These are only beneficial for low level activity e.g., walking, running. It's very hard to infer high level activities [67] e.g., recognizing a person having a coffee is approximately impossible if recognized via shallow features. Conventional PR approaches have requirement of large amount of labeled data in order to train the model but practically mostly the activities are unlabeled in real time. Similarly, many other statistical data are being leant by existing PR model but the actual data in real life is un continuous stream which needs robust , online , prompt learning. To overcome the issue of PR as shown in Fig 3-1, deep learning plays its role as shown in Fig 2-2.

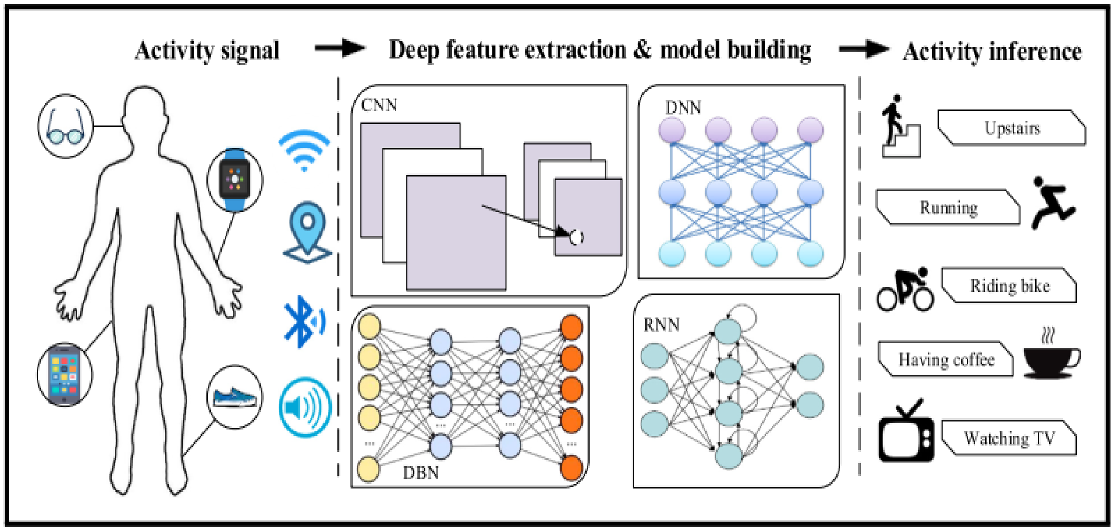

Figure 2-2 Deep Learning Approach [42]

Deep Learning is when fed with large unlabeled data, deep generative models were able to exploit for model training [24]. Thus, from above explanation we come to point that we will proceed with Deep learning methods for Human activity recognition via IMU time series data. After its fed to the respective deep learning model results would be reduced for further interpretations. Fig 2-3 shows categorized graphical view w.r.t human activity recognition.

Figure 2-3 Deep learning Categories with respect to HAR [42]

Following are some key points which make it a choice for Human activity recognition

1) One of the most important features of deep learning is layer by layer a hierarchal structure of deep model. Thus, it enables them to learn from simple to complex features. Similarly, deep models were able to learn descriptive features from resources like GPUs. So, this ability to learn complex features via resources provided makes it best for HAR w.r.t accuracy and analysis of multimodal sensor data.
2) Structure of Deep neural networks are diverse which could encide features from multiple sources e.g., CNN are able to capture local dependency of multimodal data similarly, translational invariance at local goes to accurate recognition[60]
3) Deep learning algorithm can be attached and detached making a hybrid model based on respective need e.g., deep transfer learning[3] , deep active learning [53]etc.

2.2 Selection of Deep Learning Model

Following are methods of deep learning available in recent research in field of Human activity recognition given in Fig 3-4. Out of these following available methods we would choose deep model best suited for our system and justification will be given.

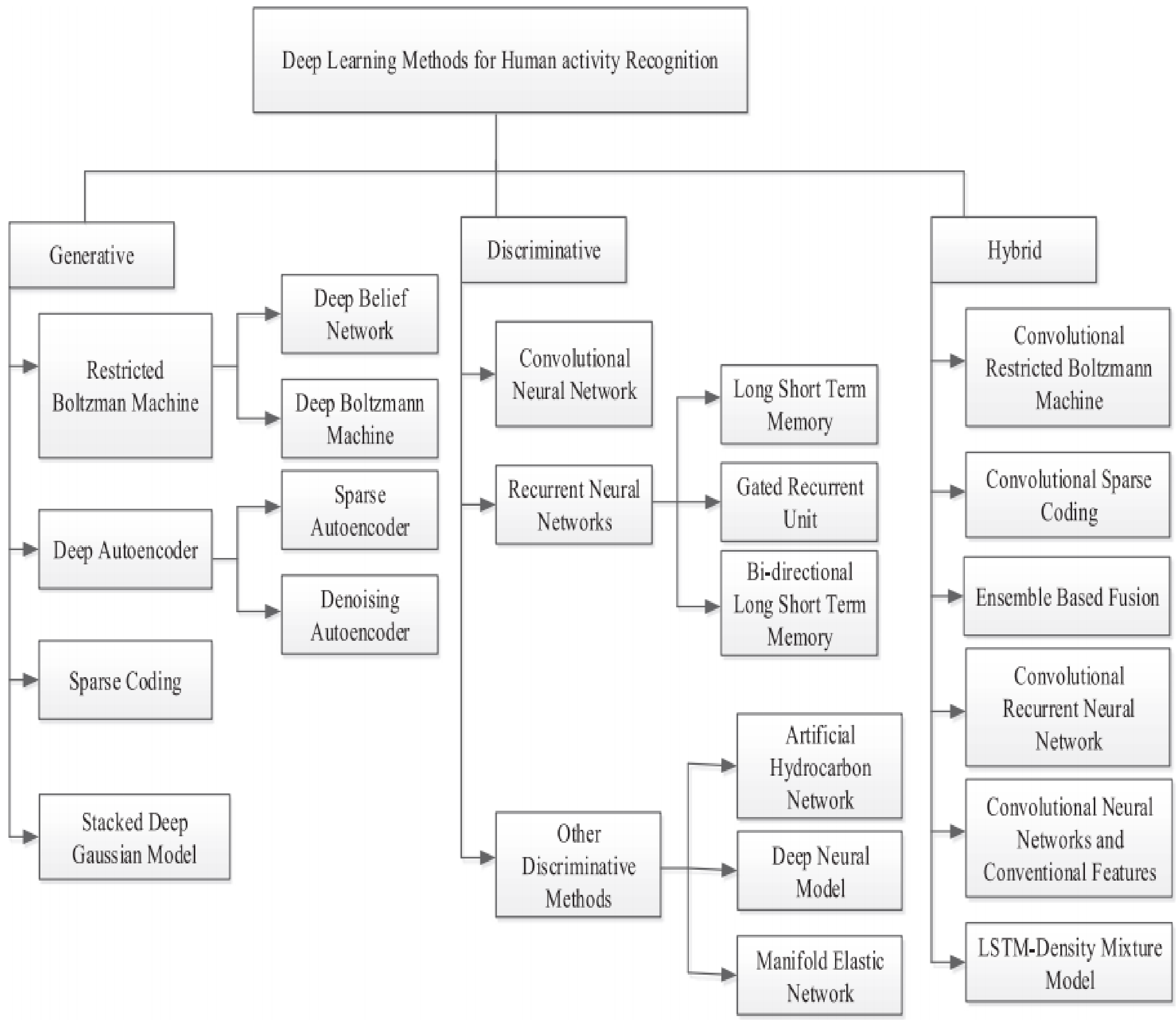

Figure 2-4 HAR Recent Algorithms [44]

From the above models of deep learning mostly used in human activity recognition, would use Convolutional Neural Network. CNN basically consist up of Deep neural network. These are interconnected with each other. CNN basically performs on raw data. I am composed up of convolutional, pooling and fully connected layer. Basically, it gathers feature with different size of kernel and strides. Later on, its pooled w.r.t features so that it could be source if reduction to reduce connections between convolution layer with pooling layer. These pooling layers make network translational invar rent to different distortions and changes by reducing features maps, parameters. In the following given table , CNN is explained and reason for selecting this model of deep learning for out Humana activity recognition.

Table 2-1 Comparison CNN

Deep Model	Strength & Weakness	Recent Application	References of literature CNN used
CNN	Widely used in DL	Relationship reduction between exercise and sleeping patterns	[45,46,47,48,49,50,51,52]
	Many training strategies proposed in it	Automatic sports pain detection	
	Automatically learns features from raw sensor data	Personal Activities tracking	
	CNN sensory data orientation with activity details are invariant	Estimation of energy exerted in any particular task	
	Can be used to model time dependency.	Human activity recognition	

Similarly following are the other deep learning model being used in literature for HAR.

Table 2-2 Deep Learning Popular Methods

Sr	Name	Explanation
1	DNN	Fully connected Deep network, ANN with deep layers [41]
2	CNN	Convolutional Neural Network [42]
3	RNN	Recurrent Neural Network [42]
4	DBN/RBM	Restricted Boltzmann machine or Deep belief network [42]
5	SAE	Stacked Autoencoder [33]
6	HYBRID	Different combo of different deep models [43]

2.3 Deep Learning Implementation framework:

Deep learning is now an important part of research. Different hardware and software for implementations have been developed until now. These hardware and software have enabled researchers for high performance computing for HAR. As these frameworks includes open source and some are IP based developed by many organizations for cutting edge tech development. Graphic cards like Nvidia are playing important role in development of GPUs and different types of processors that could help to accelerate performance and learning increased capabilities in deep learning methods. Following libraries were used recognition via CNN on Google colab for training on data set.

2.3.1 TensorFlow:

TensorFlow is basically an open-source platform framework created by Google Research Department. It was basically for numerical computation via data flow of graph. It's been widely used by research community for deep leaning as its very user friendly and flexible for many algorithms , even can run on mobile devices. It is capable of providing support from low level to high level network training via multiple GPU and provides consistent parameter upgradation.

2.3.2 Keras:

Keras was basically pre-requisite for TensorFlow and Theano in deep learning for coding in python language. It has ability of quick implementation of high level NNs of deep learning. Mainly it's a support for Theano and tensor flow. This platform allows to be modular, user friendly platform via python language.

2.3.3 Pandas:

Pandas is basically a powerful tool. It is basically used for data analysis and its manipulation of data in respective data frames. It's basically written in python and C. It can be used for various task like filtering of data, segmentation, segregating of data.

2.3.4 Numpy:

Numpy is basically library of python used to deal with arrays. It can be used for variety of mathematical operations w.r.t arrays. It's a good tool of python used in data structures that guarantee efficient calculation with arrays. It has high level functions of mathematics w.r.t matrices and arrays.

2.3.5 Matpolip:

Matplotlib is commonly used library which is a comprehensive library. It basically used for creating static, visualizations and animated in Python.

2.3.6 Sklearn:

Sklearn is a robust library use for machine learning in python. It contains efficient tools for ML and statistical modeling e.g., regression, clustering etc. It can be helpful in dimension reduction.

2.3.7 Pydrive:

Pydrive is basically a high-level python based google drive API. It helps for upload, delete, download of files on google drive.

2.4 Platform for HAR:

In the following activity recognition of human Google Collaboratory was used for training on data set. It is best suited platform from learning to research on different ML algorithms. Rest Raspberry Pi was used as computing platform in real-time prediction.

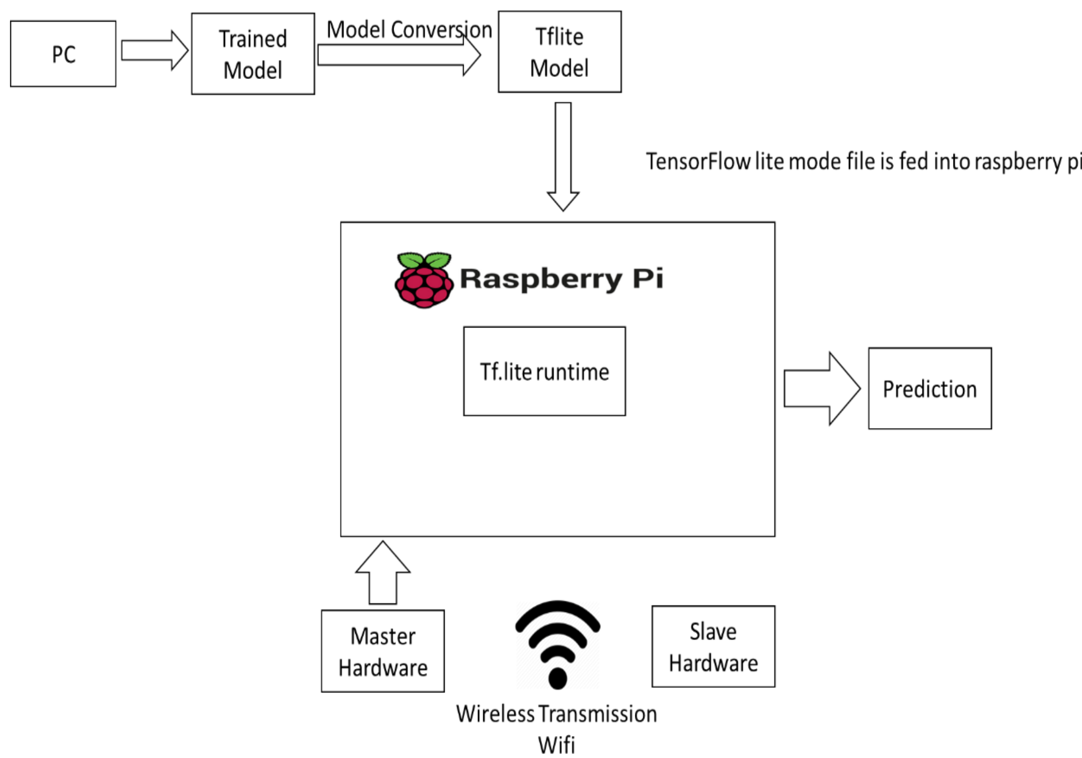

Figure 2-5 Raspberry pi Workflow

2.5 Sensory for Human activity Recognition : -

Generally, in literature, there are mainly two types of methods for human activity recognition [42]

1) Vision Based System
2) Sensor Based system

Similarly, sensor-based system includes

1)Wearable

2)Device Bound

3) Device Free

Whereas visual sensor includes the use of camera for video or imaging capture for identification of any object. AS there are many problems in visual based sensor among its privacy was most important factor as it's not possible to use camera at every location. With that there was also a main issue of computing of the computer vision-based data. While these wearable sensors used to perform same as task as video with flexibility in it.

Hence, IMU sensor in wearable condition used in experiment. ESP32 is used to receive signal form slave ESP32 to process at Raspberry Pi for activity recognition.

Chapter 3: Algorithm Explanation:

Importing libraries and mounting google Drive:

Initially, we would be mounting google drive with our Google Colab setup for directly reading dataset from google drive and later on save file in directory. For those we would be importing libraries of google drive via google colab . Similarly, after mounting the google drive library, we would be setting the directory path. Once we complete this basic setting w.r.t google colab and drive , we would then proceed for installing libraries of Deep learning .Mainly following are the libraries which would be used in our algorithm

1) TensorFlow
2) Keras
3) Pandas
4) NumPy
5) Matplotlib
6) Sklearn
7) Pydrive

Once all libraries are installed, our Google Collaboratory is ready for coding for HAR.

3.1.1 Reading Data File:

At the initiation of the algorithm, we would be reading data files having sensor values and labels. The main target for this step is to adjust the dataset according to list for ease of processing.

3.1.2 List Creation:

Now after the data has been read from dataset file , it would further proceed for list creation. As the file contain 12 sensor values so each one of them is further processed to assign it to 1 label. As list is created via elements inside square bracket and each of the element is separated by comas.

3.1.3 Initialization of data frame

Once list of data is created then data is prepared for framing. As the list is processed to columns and shape of data frame is determined. In our case the shape of data frame is (178862,14) . Similarly, as a sample output to visualize data frame data.head(10) results are listed in table below.

Table 3-1 Processed Data in Dataframes

index	user	activity	ax1	ay1	az1	gx1	gy1	gz1	ax2	ay2	az2	gx2	gy2	gz2
0	1	Walk	62	2564	4118	18	-2	-55	-1936	-1652	1198	2	-11	11
1	1	walk	62	2564	4118	18	-2	-55	-1934	-1402	1190	2	-11	12
2	1	walk	558	2948	3482	18	-2	-55	-1934	-1402	1190	2	-11	12
3	1	walk	558	2948	3482	18	-2	-55	-1609	-966	1104	1	-12	13
4	1	walk	558	2948	3482	18	-2	-55	-1081	-503	933	0	-12	14
5	1	walk	2294	3612	2113	19	-1	-55	-1081	-503	933	0	-12	14
6	1	walk	2294	3612	2113	19	-1	-55	-554	-191	750	0	-12	15
7	1	walk	3085	3666	1558	20	-1	-54	-554	-191	750	0	-12	15
8	1	walk	3772	3592	958	20	0	-54	-554	-191	750	0	-12	15

3.1.4 Counter Checking of Non-Null values:

After the initialization of data frames and listing creation , the next step is to counter check the data information as counter check that either all are non-null or not. For this data.info() function was used with data.isnull().sum(). This checking on data frames output is shown below in table.

Table 3-2 Non-Null Values checking

#	**Column**	**Non-Null Count**	**Data Type**
0	User	178862 non-null	object
1	Activity	178862 non-null	object
2	ax1	178862 non-null	object

3	ay1	178862 non-null	object
4	az1	178862 non-null	object
5	gx1	178862 non-null	object
6	gy1	178862 non-null	object
7	gz1	178862 non-null	Object
8	ax2	178862 non-null	Object
9	ay2	178862 non-null	Object
10	az2	178862 non-null	Object
11	gx2	178862 non-null	Object
12	gy2	178862 non-null	Object
13	gz2	178862 non-null	Object

3.1.5 Data Dropping:

Before proceeding forward, data is refined so that it cannot create problem in classification e.g., we could remove the user named column as it's of no need. As data is in pre-processing phase where unnecessary data is removed.

Table 3-3 Sample Walking data without User ID column

index	activity	ax1	ay1	az1	gx1	gy1	gz1	ax2	ay2	az2	gx2	gy2	gz2
0	walk	62.0	2564.0	4118.0	18.0	-2.0	-55.0	-1936.0	-1652.0	1198.0	2.0	-11.0	11.0
1	walk	62.0	2564.0	4118.0	18.0	-2.0	-55.0	-1934.0	-1402.0	1190.0	2.0	-11.0	12.0
2	walk	558.0	2948.0	3482.0	18.0	-2.0	-55.0	-1934.0	-1402.0	1190.0	2.0	-11.0	12.0
3	walk	558.0	2948.0	3482.0	18.0	-2.0	-55.0	-1609.0	-966.0	1104.0	1.0	-12.0	13.0

3.1.6 Plotting activities with Window size:

As now we would be plotting our all activities which were in pipeline in variables of accelerometer. With that Fs for plotting window size is also used. With respect to time domain following results are plotted from given data.

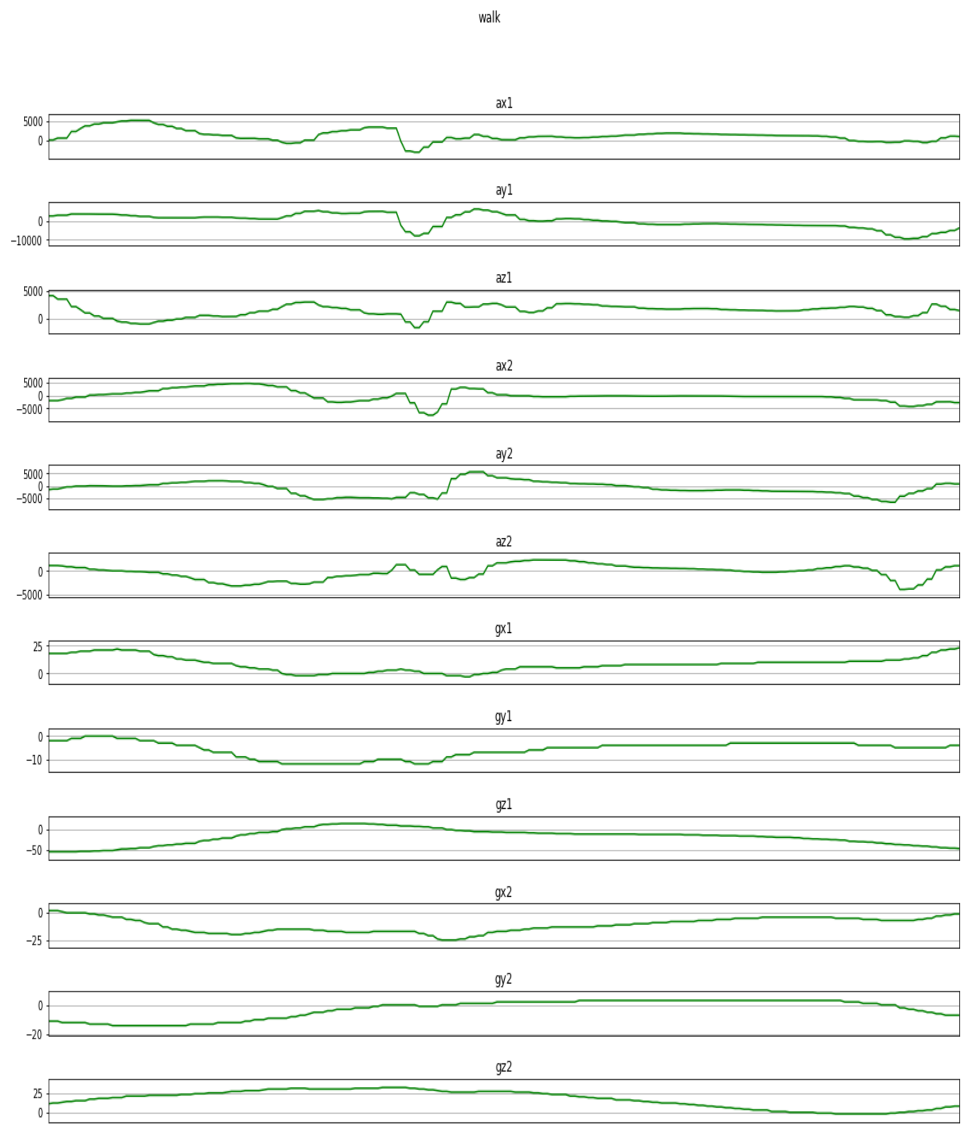

Figure 3-1 Accelerometer Walking Activity Plot

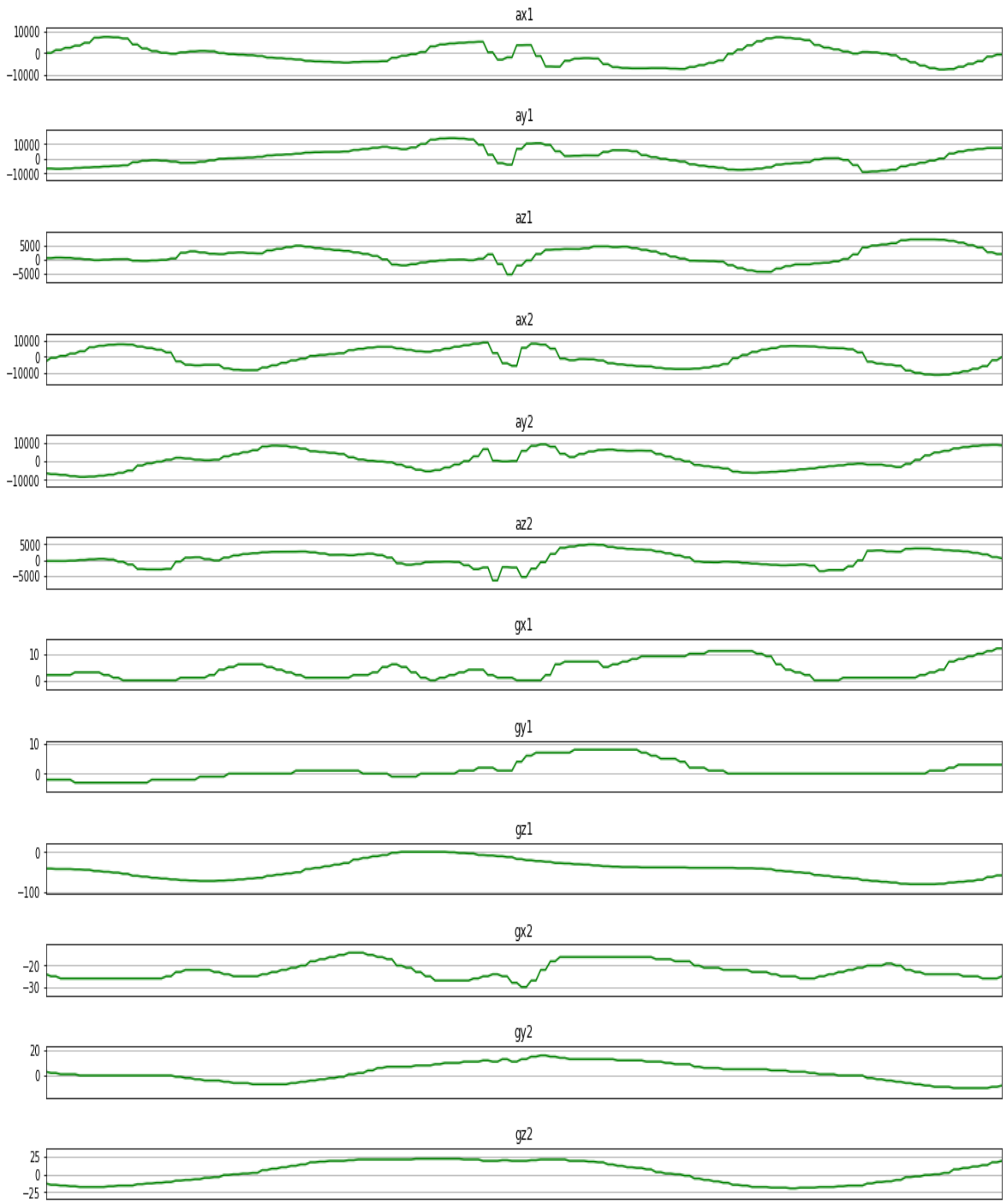

Figure 3-2 Running Accelerometer

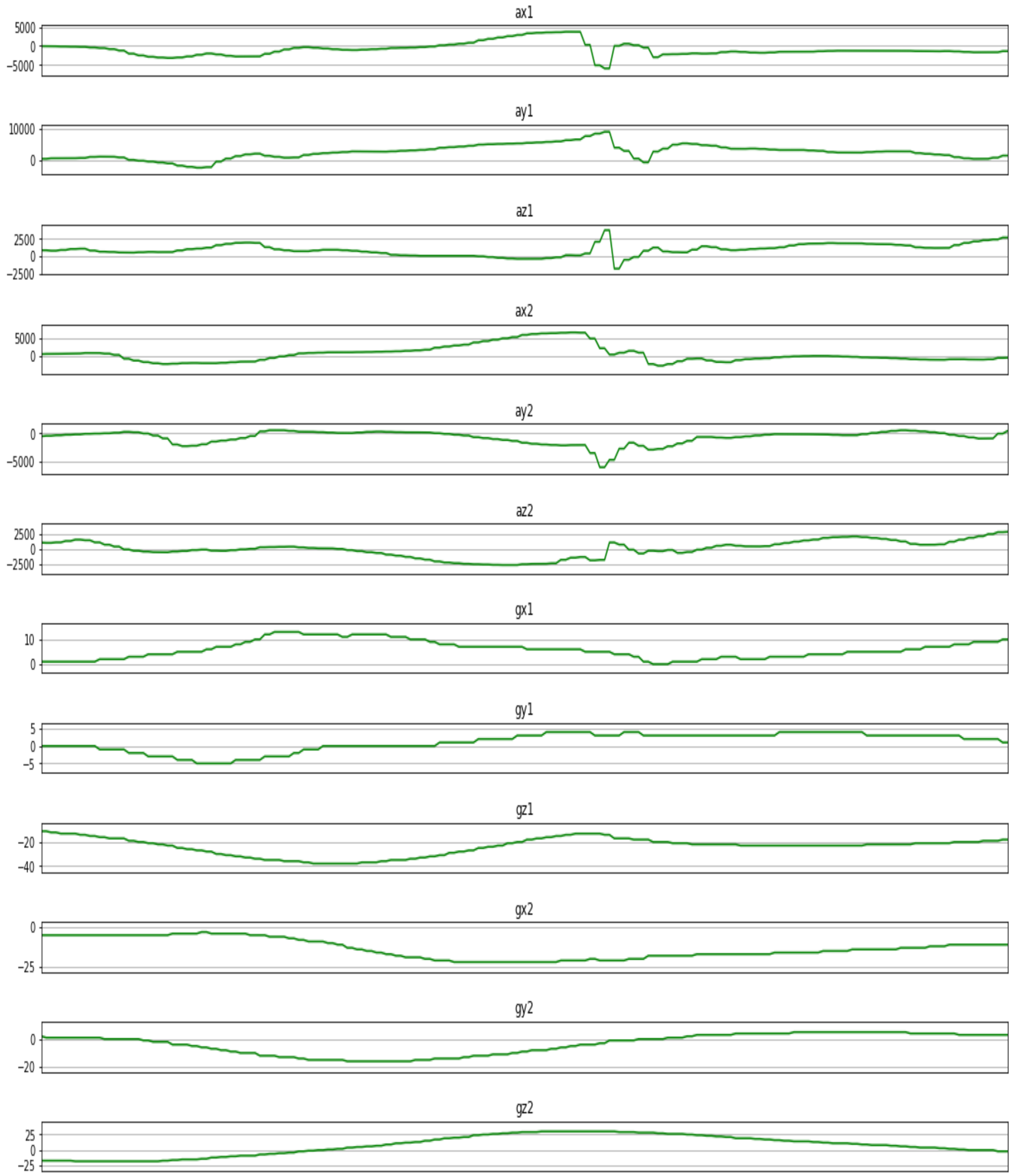

Figure 3-3 Stair up Accelerometer Date

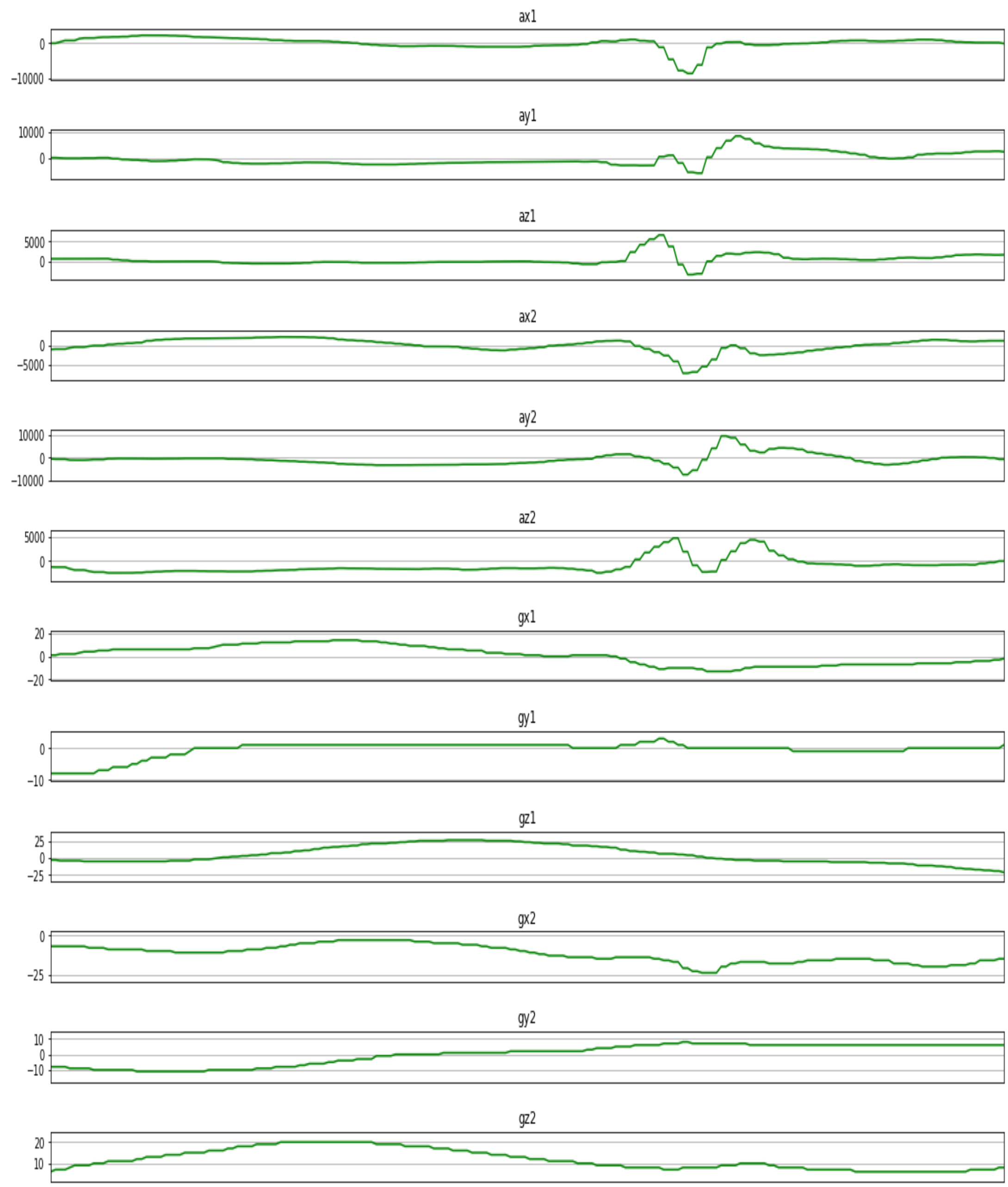

Figure 3-4 Stair Down Accelerometer Data

3.1.7 Normalization of sensor values:

Normalization is processed for incoming sensor values because it's a pre-processing method used to make our data standardized. We basically map different type of source of data to same common range so that when it is feed to CNN, results should be more accurate. If the incoming data is not normalized, then to can create problem while training the CNN network. As there is a chance that CNN may learn other our requirement because of not normalizing our data. So, it is considered a good practice to normalize the data before it is feed into the network. For the process of normalization.

For this purpose, standard scaler, fit_transform() functions are used. In picture below, we can see normalized data.

Table 3-4 Normalized Data for training

	ax1	ay1	az1	gx1	gy1	gz1	ax2	ay2	az2	gx2	gy2	gz2	label
0	-0.012600	0.526021	1.490694	0.783285	-0.687015	-1.486339	-0.504650	-0.451046	0.571672	1.484309	-1.875517	0.027338	3
1	-0.012600	0.526021	1.490694	0.783285	-0.687015	-1.486339	-0.504127	-0.377856	0.567519	1.484309	-1.875517	0.063824	3
2	0.128329	0.611093	1.185823	0.783285	-0.687015	-1.486339	-0.504127	-0.377856	0.567519	1.484309	-1.875517	0.063824	3
3	0.128329	0.611093	1.185823	0.783285	-0.687015	-1.486339	-0.419174	-0.250213	0.522869	1.376529	-2.050644	0.100310	3
4	0.128329	0.611093	1.185823	0.783285	-0.687015	-1.486339	-0.281158	-0.114664	0.434087	1.268750	-2.050644	0.136797	3
...	...	...	...	...	...	...	...	...	...	...	...	...	...
107911	-0.146142	0.452027	0.512806	0.172374	-0.372094	0.300139	-0.422834	1.472976	0.534291	-0.455720	-0.299380	-0.228064	1
107912	-0.146142	0.452027	0.512806	0.172374	-0.372094	0.300139	-0.398785	1.398615	0.417992	-0.455720	-0.299380	-0.264551	1
107913	-0.296164	0.176210	0.273607	0.172374	-0.372094	0.300139	-0.398785	1.398615	0.417992	-0.455720	-0.299380	-0.264551	1
107914	-0.296164	0.176210	0.273607	0.172374	-0.372094	0.300139	-0.405059	1.116101	0.447586	-0.455720	-0.299380	-0.264551	1
107915	-0.426296	-0.095618	0.056937	0.172374	-0.372094	0.300139	-0.405059	1.116101	0.447586	-0.455720	-0.299380	-0.264551	1

3.1.8 Labeling data:

Now days is labeled via Sklearn library. In labeling each of the activity is assigned with a label which would be indication of recognizing certain activity upon successful recognizing CNN would predict based on labels assigned to each activity e.g., as a sample shown in fig where ‘3’ as a label is characterized to walking activity.

Table 3-5 Data Framing for CNN model

activity	ax1	ay1	az1	gx1	gy1	gz1	ax2	ay2	az2	gx2	gy2	gz2	label
walk	62.0	2564.0	4118.0	18.0	-2.0	-55.0	-1936	-1652	1198.0	2.0	-11.0	11.0	3
walk	62.0	2564.0	4118.0	18.0	-2.0	-55.0	-1934	-1402	1190.0	2.0	-11.0	12.0	3
walk	558.0	2948.0	3482.0	18.0	-2.0	-55.0	-1934	-1402	1190.0	2.0	-11.0	12.0	3
walk	558.0	2948.0	3482.0	18.0	-2.0	-55.0	-1609	-966	1104.0	1.0	-12.0	13.0	3

3.1.9 Framing Data for CNN model:

Table 3-5 Data Framing for CNN model

frame size as

Fs=20

Fs*15=300

Where hope size =Fs*2=40

Similarly, after defining frame size , data is fed into for loop for data points conversion to frames. After its done , we will retrieve most often used labels in this segment. After identification we will bring segments into better shape

3.1.10 Splitting of Data for Testing and training:

After the data points are converted to respective frame sizes , next step is to split this data into testing and training data. Thus, shape for X-train.shape is (2152,300) while X-test.shape (539,300,12). Later on, its reshaped to X_train[0].shape and X-test[0].shape both to (300,12,1). Further its fed into CNN model architecture.

3.1.11 Initiation of CNN architecture

After the successfully able to split our data to test data and train data , we would proceed for CNN structure where we would be able to use different hidden layers, dropouts , Flatten , Dense , softmax layers. As we would not be using conventional way of introducing pooling layers. Though pooling layers reduce training time but at the same time it decreases the accuracy.

a. Input:

The data having 12 features, or six axis accelerometer data is fed into network.

b. Convolution:

Convolution algorithm is deployed so that matrix multiplication can be avoided as previously used in traditional methods. The kernel sizes were 16,64 .

c. Dropout:

This layer mainly used to eliminate overfitting in covnets. In this experiment the dropout layers used were 0.1,0.2,0.5 respectively.

d. Flatten:

Following layer is sued to convert the 2D array of convolved into single long linear vector. So that we can model our input and build network passing through each neuron of covnet effectively. It is making the fully connected layer.

e. Activation function:

It is a rectified linear unit used in covnets. Activation Function basically defines weighted sum of inputs are transformed to output via node from a network of nodes. In the following experiment , the activation function used is softmax.

f. Output:

As the output layer of fully connected layers is basically fully positioned with softmax layer. The softmax function basically computes each probability at each node , the node with higher probability is considered as prediction. Following Detailed Fig shows CNN structure used in this experiment.

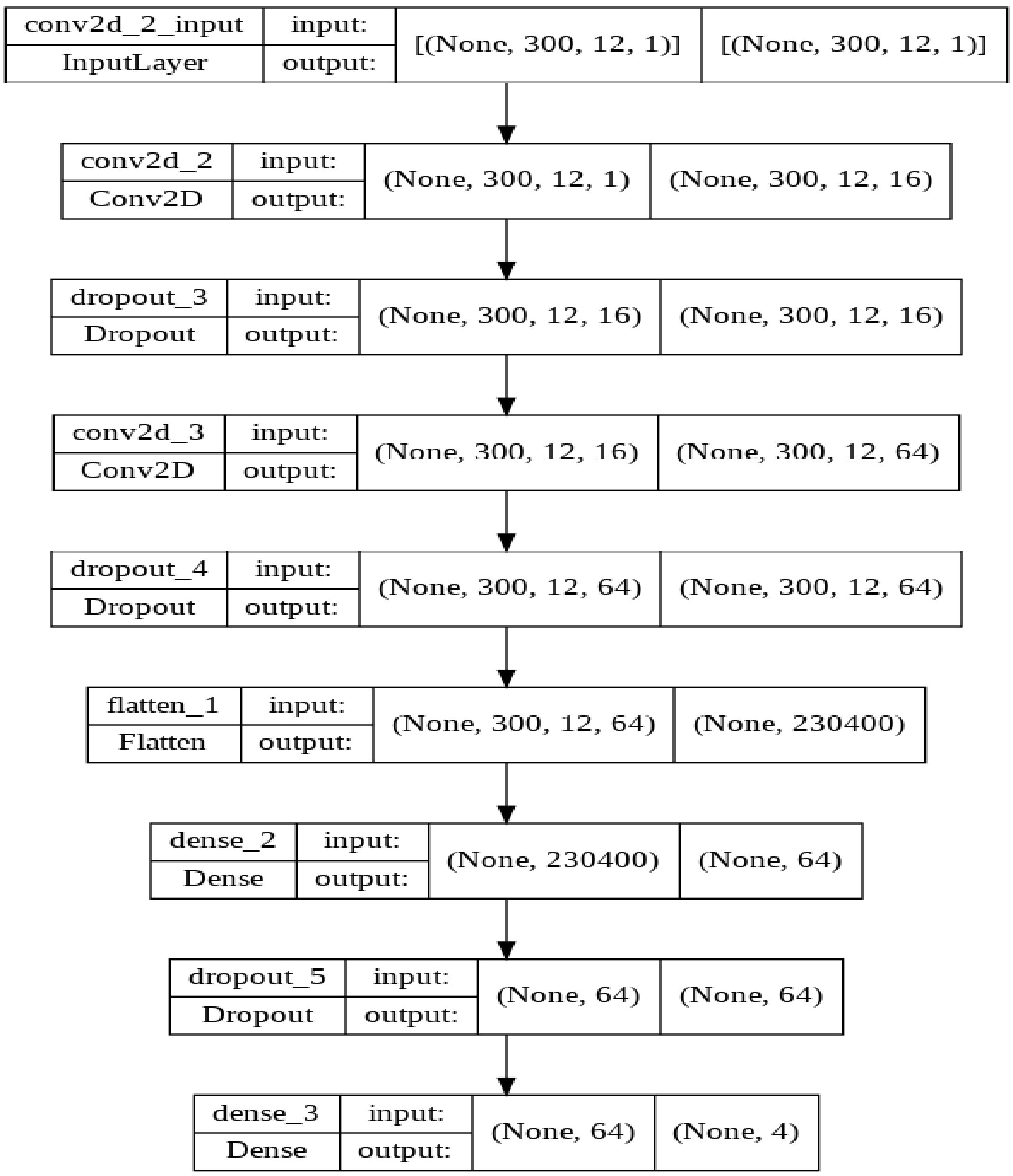

Figure 3-5 CNN Structure

3.1.12 Training of CNN architecture

In the following CNN-2D architecture, we used optimizer 'Adam'. With the respective learning rate of 0.0002. The loss function was selected as Sparse Categorical Cross entropy. Similarly, 10 epochs were used for this experiment.

Chapter 4: Experimental Setup

4.1 Hardware Setup for Data collection & Training

For the respective experiment, it is consisting up of two parts hardware and software. Till now software part w.r.t training has been explained. In order to move towards implementation, we need to describe about biases of training code and data set collection mechanism, which was fed into CNN for classification for HAR,

The following algorithm is designed w.r.t WISDM dataset as standard. As its used widely by researchers. With the approach of taking WISDM as a standard to design algorithm, its necessary that dataset should be collected following WISDM as standard. The recording of data set was done as per standard of WISDM [53], described in Table.

Table 4-1 WISDM Standard

Field	Description
Subject id	Type: Numeric Value Description: this is the id/serial of the participant recording his data
Activity Code	Type: Activity Name Description: this is the name of the activity being recorded i.e., walking, running etc.
Ax	x-axis value of accelerometer
Ay	y-axis value of accelerometer
Az	z-axis value of accelerometer
Gx	x-axis value of gyroscope
Gy	y- axis value of gyroscope
Gz	z- axis value of gyroscope

Following the approach of WISDM [53] a thesis work on Data set collection via IMU sensors is used. In the following study [54], two IMU sensors were used which were attached to leg of different subjects for dataset collections. As following Dataset from study [54] is recorded which is later on also processed. The dataset recorded in the study [54] was w.r.t walking, jogging, stair climbing up, stairs climbing down. In following study [54], 16 subjects' data were recorded in these four different activities which our algorithm would also be classifying. Hardware used for collection of datasets shown in figure.

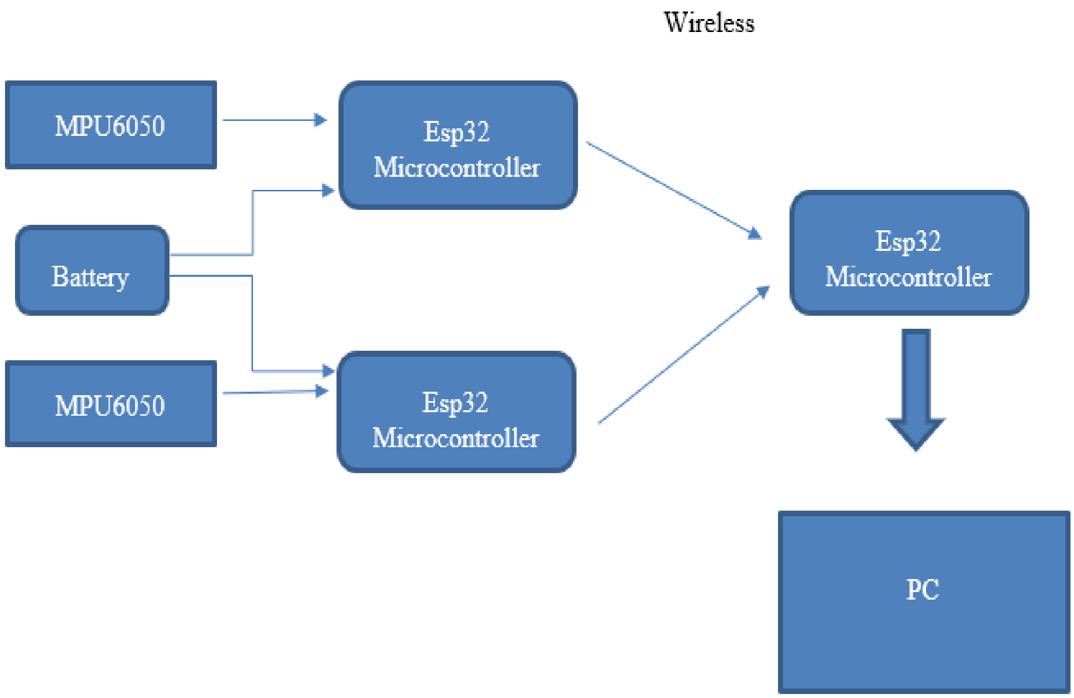

Figure 4-1 Dataset collection layout [54]

Both MPU6050 are attached to subject body. The data of IMU sensor is transmitted continuously via ESP32 microcontroller to another master Esp32 which process following signals on operating system. As Both ESP32 at slave end transmit data over Wi-Fi wirelessly. Which is then received by master ESP32. Later on, this Master Esp32 transmit to Operating system where the signals are pre-processed before fed into CNN. Following Figure clearly shows hardware setup used in study [54] for data collection.

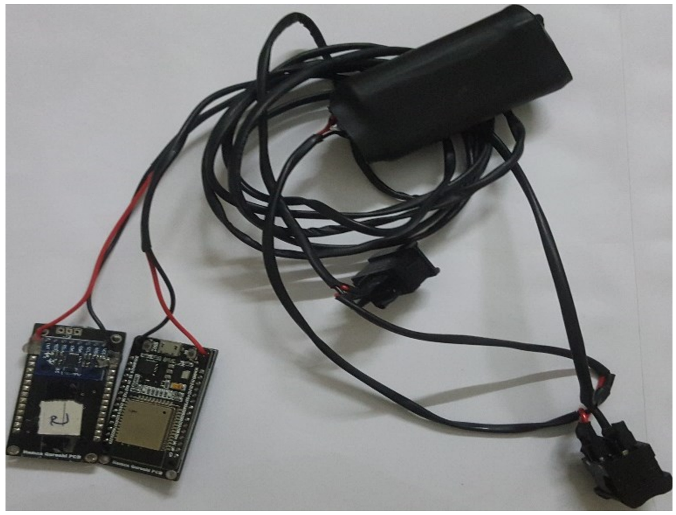

Figure 4-2 Data collection Hardware [54]

Data set collected via following hardware is further processed to convert in such a format with labeling which could play role in accuracy later on when dataset is fed into CNN via frames. Dataset after pre-processing would in following format as shown a sample in table

Table 4-2 Processed Data

A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	walk	62	2564	4118	18	-2	-55	-1936	-1652	1198	2	-11	11
1	walk	62	2564	4118	18	-2	-55	-1934	-1402	1190	2	-11	12
1	walk	558	2948	3482	18	-2	-55	-1934	-1402	1190	2	-11	12
1	walk	558	2948	3482	18	-2	-55	-1609	-966	1104	1	-12	13
1	walk	558	2948	3482	18	-2	-55	-1081	-503	933	0	-12	14
1	walk	2294	3612	2113	19	-1	-55	-1081	-503	933	0	-12	14
1	walk	2294	3612	2113	19	-1	-55	-554	-191	750	0	-12	15
1	walk	3085	3666	1558	20	-1	-54	-554	-191	750	0	-12	15
1	walk	3772	3592	958	20	0	-54	-554	-191	750	0	-12	15
1	walk	3772	3592	958	20	0	-54	229	-39	400	-1	-13	17
1	walk	4258	3532	407	21	0	-53	229	-39	400	-1	-13	17
1	walk	4258	3532	407	21	0	-53	407	-91	247	-2	-13	18
1	walk	4571	3505	-22	21	0	-52	407	-91	247	-2	-13	18
1	walk	4571	3505	-22	21	0	-52	546	-171	145	-3	-13	18
1	walk	4571	3505	-22	21	0	-52	686	-222	62	-4	-14	19
1	walk	4828	3347	-528	22	-1	-50	686	-222	62	-4	-14	19

4.2 Different dataset results:

Data set was collected at different location first. Based on that a final version of data set was collected. There was total 3 type of data sets which were recorded on 3 different locations. First location was at right arm, second location was right-left arm, and third location was right leg.

Based on training, testing results we would select dataset which could give good results. Among 3 different types of configurations, Right Leg configuration was selected. As shown in table of accuracies. Similarly, in depth the result of right leg is discussed in Results and Discussion section.

Table 4-3 Dataset Accuracy

Dataset Name	**Accuracy (%)**
Right Arm	81 %
Right-Left Arm	95 %
Right Leg	97.58 %

Right Arm:

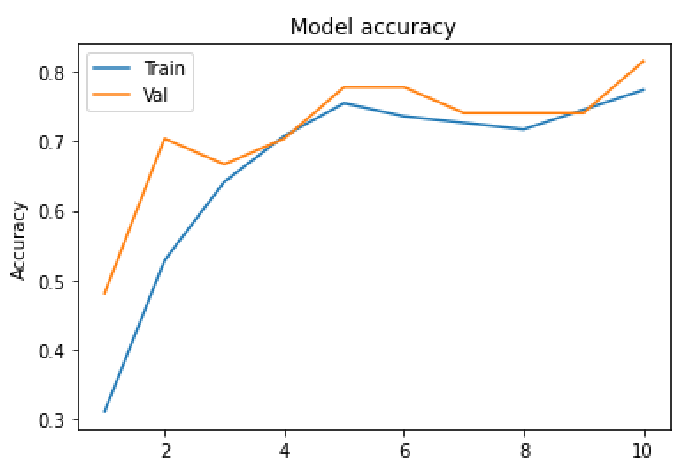

Figure 4-3 Model Accuracy Right Arm

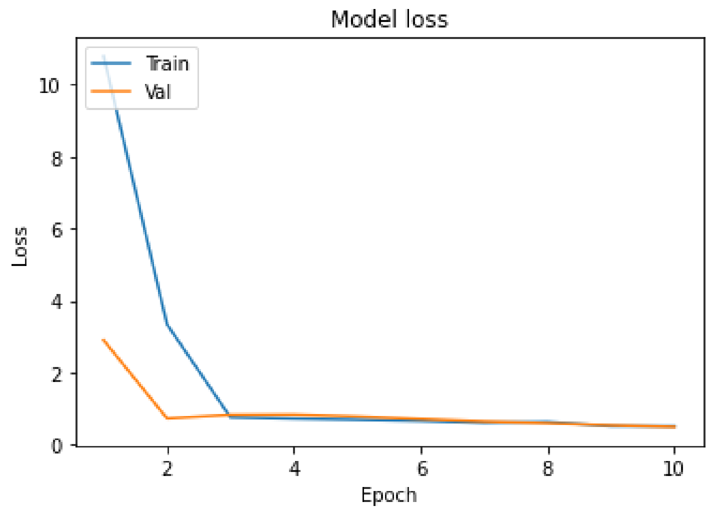

Figure 4-4 Model Loss Right Arm

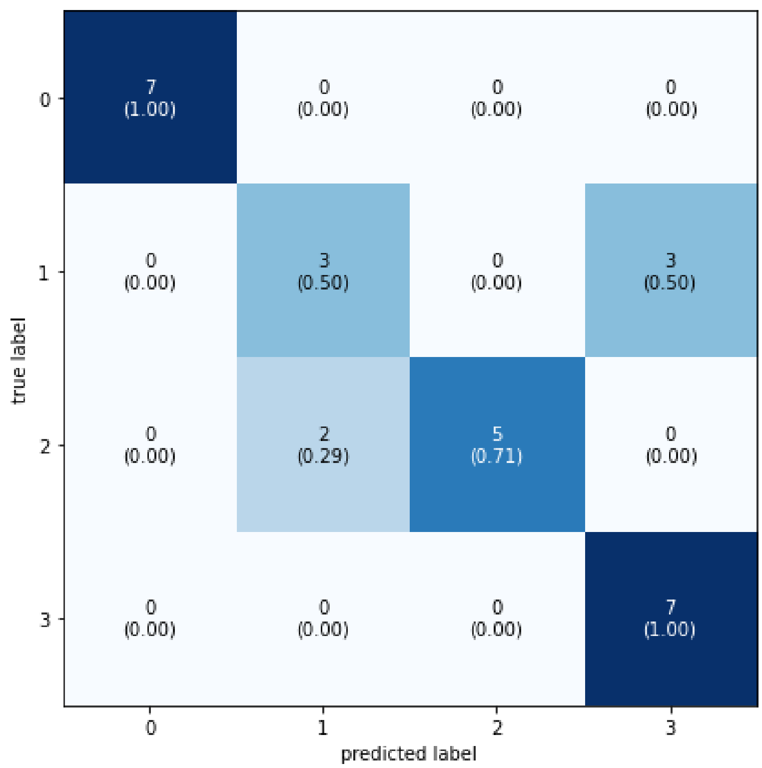

Figure 4-5 Confusion Matrix Right Arm

Right-Left Arm:

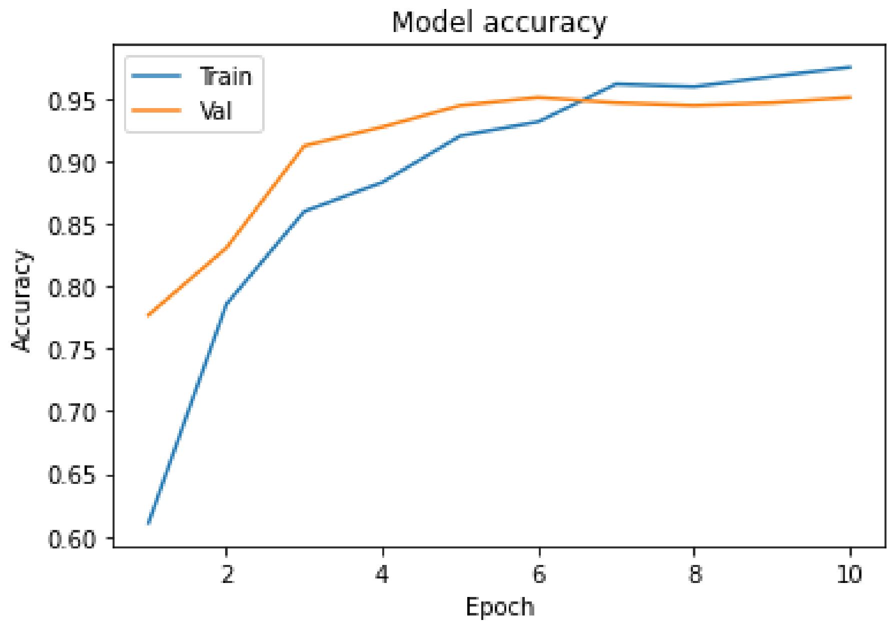

Figure 4-6 Model Accuracy Right Left Arm

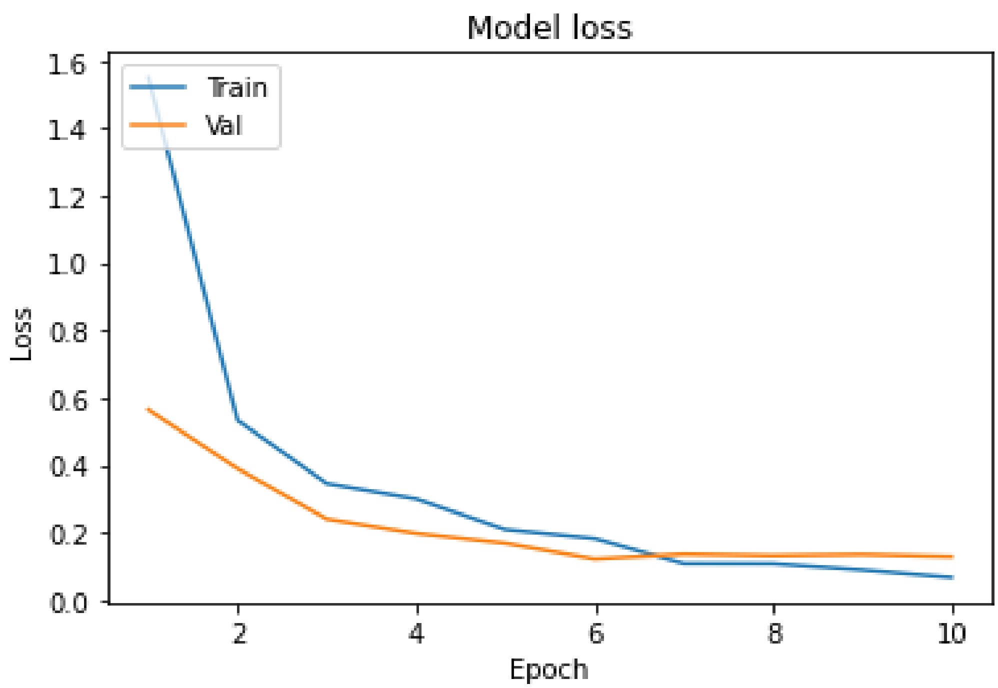

Figure 4-7 Model Loss Right Left Arm

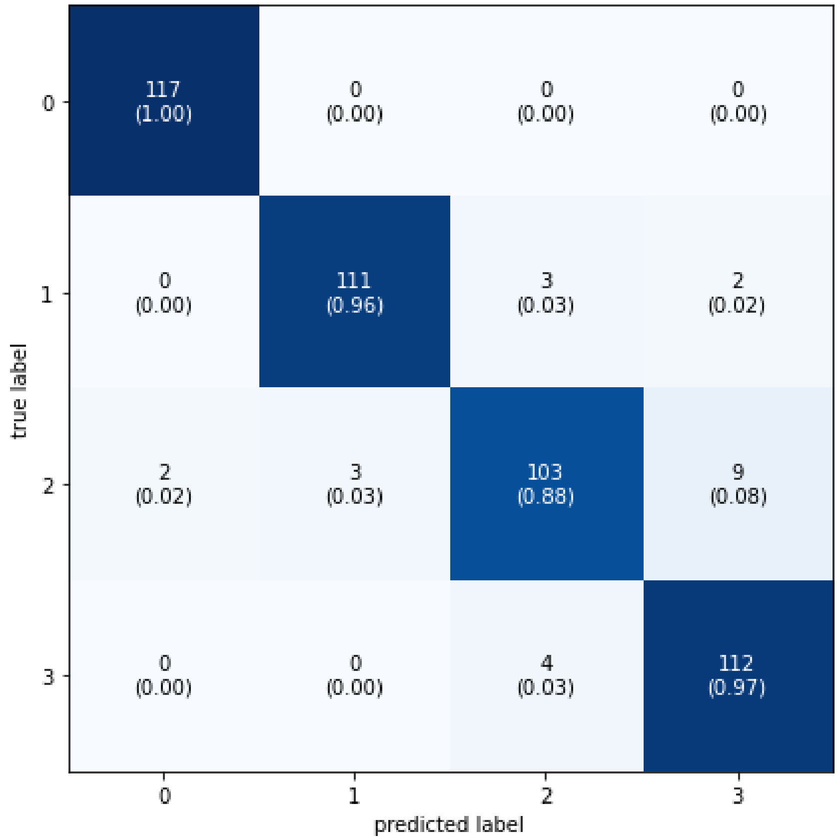

Figure 4-8 Confusion Matrix Right Left Arm

4.3 Hardware setup for Testing

After successful data recording and pre-processing, data is ready to be fed into CNN algorithm for testing or classification of Human activity recognition. Now the hardware used in research article [54] for data se collection, would be used again but for this time for classification. As slave MPU6050 would be used again with same architecture but master MPU6050 would be connected to Raspberry pi. Now Raspberry pi would be used to receive the data from mater MPU6050. After the data starts receiving in Realtime, Prediction would be caried out for HAR at raspberry pi. Once an activity is predicted or classified, further signal would be sent to motor driver controller to operate prosthetic knee stepper motor for opening and closing of joints according to run, walk or climbing. A brief graphical view is given for classification approach figure

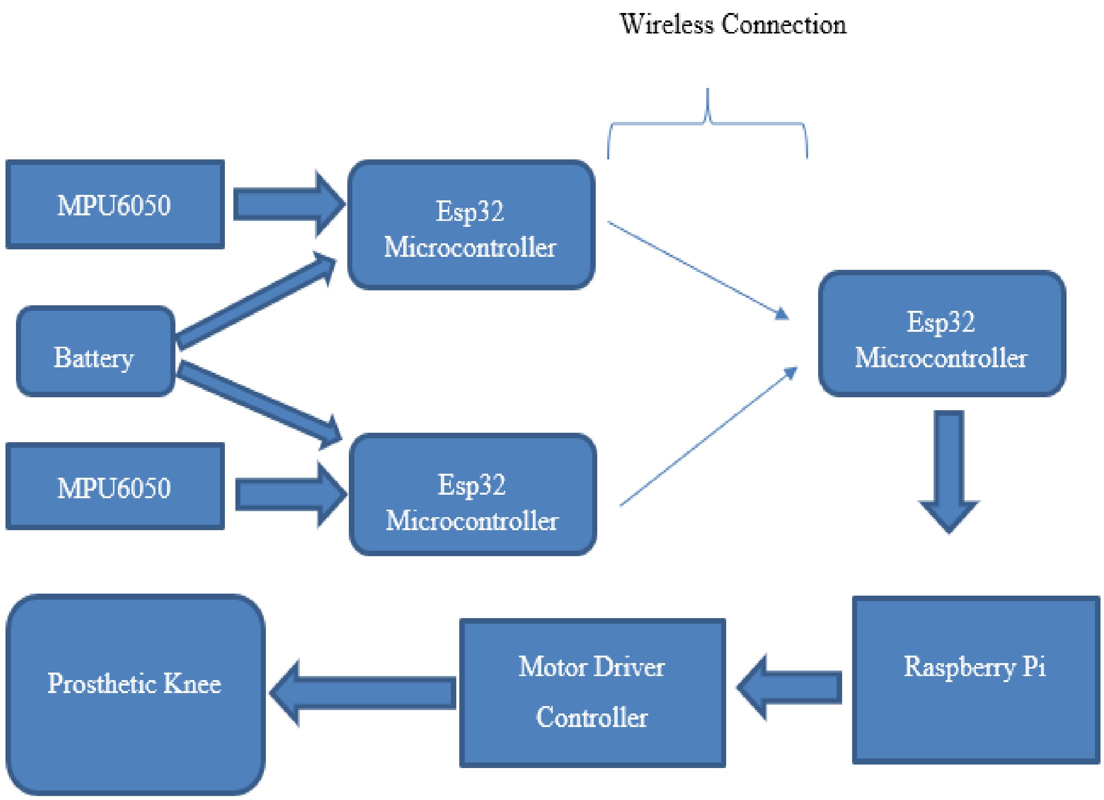

Figure 4-9 Experimental Setup Layout

Now after classification results, based on motor driver hardware, values are communicated to prosthetic knee. As stepper motors are attached on prosthetic knee, following set of combinations were classified w.r.t classification activities. Based on previous work [53], switches were previously used as combination if 3 different states , which results in 8 different combinations. Now firstly , we will have live streaming of data being receive data raspberry pi. At raspberry pi prediction is made. Further , a motor driver is attached as used in [53] with raspberry pi. Based on prediction raspberry pi gives signal to motor driver to control stepper motor with respective adjustment. Following Table 4-3, shows different combinations states which are assigned with different activities.

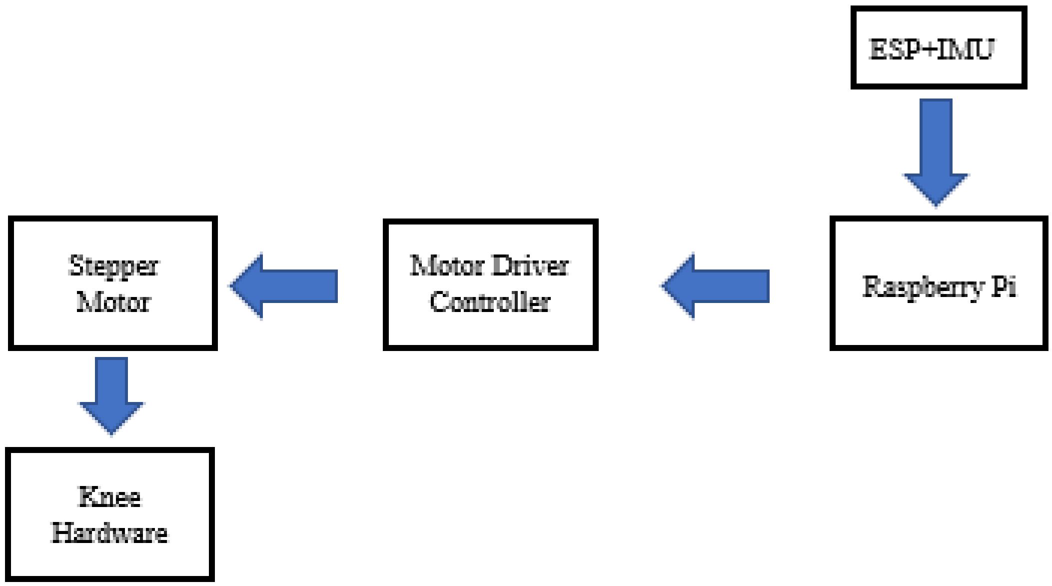

Figure 4-10 General Layout

Table 4-4 Raspberry O/P States

Output State	**Input State**		
Stop	0	0	0
Walk	1	0	0
Stair Up	0	1	0
Run	1	1	0
Stair Down	0	0	1
Default Position (current position)	1	0	1
Lock	0	1	1
Reset	1	1	1

4.4 Key Hardware Components

a. Raspberry Pi:

Raspberry Pi 4 8gb is latest variant of Pi family. It offers increase in speed of processor, multimedia a performance, more memory and connectivity. It provides performance compared to x86 Pc system of entry level. It has 64bit quad processor which has high performance, dual display , 4K resolution. Due to such high-performance features , it requires 3.0 A USB-C power supply.

Figure 4-11 Raspberry pi [55]

b. MPU6050

This is a sensor used to track 6-axis motion of any object or body. It has 3 gyroscope axis ,3 accelerometer axis and motion processor which is digital. These three are in once package. It comes with additional feature of temperature sensor with I2C bus interface for communication.

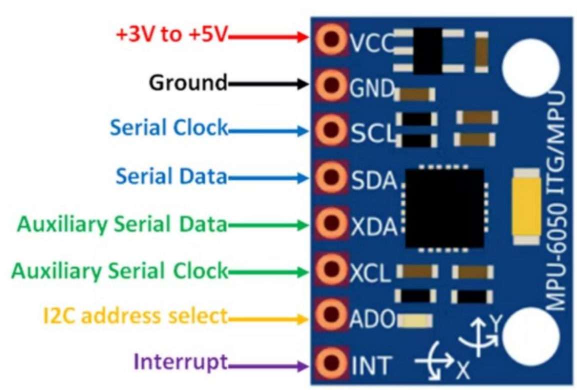

Figure 4-12 MPU6050 Sensor[55]

Chapter 5: Results, Conclusion & Future Recommendations

5.1 Results & Conclusion

The following experiment was performed in PC for training purposes and later on tested on raspberry pi . The PC specification were as Intel(R) Core(TM) i7-6700HQ CPU @ 2.60GHz 2.59 GHz. While Raspberry Pi 4 8gb was used for testing purpose of perdition of Human activity state from either 4 of state, walking, running, stair-up, stair-down. Raspberry Pi was 64bit quad-core processor which is equivalent with x86 PC system of entry level.

Results parameter would include Accuracy, precision , Recall and F1 Score. Basically, precision is related to how much accuracy of prediction was w.r.t correct recognition of Human activity recognition. Recall means how much opportunity of correct classification being under correct predicted. If precision is high , it means there is high probability if correct results being predicted.

Similarly, if there is high recall value, it means high probability of more comprehensively. F1-socre basically is used to assess balance between recall and precision. If precision and recall are similar, then F1-score would always be high. If one of the precision or recall does perform good and other poor, then still F1-socre would be poor. For it to be high it is necessary for both parameters to be high.

Now from the following literature [33][37][40][41], it was observed that most of the classification was having problem in classifying walk with run or vice versa. Like in [33] , Figure 1-20 , Figure 1-21 we can see that after training of 1D-CNN which is different from our method 2D-CNN , it was observed that confusion matrix was indicating problem in classification between walk activity in both architecture of F10 and F20 respectively. Moreover, model accuracy was achieved by the approach used in [33] was 91.32% and 92.71% for F10, F20 respectively. As compared to following study, our proposed 2D-CNN architecture not only outperforms the approach of using 1D-CNN in terms of accuracy but also indicates that 2D-CNN can perform better for very closely related activity recognition like walk. Figure 5-1, Figure 5-2 shows model accuracy and loss function on following data set as shown below.

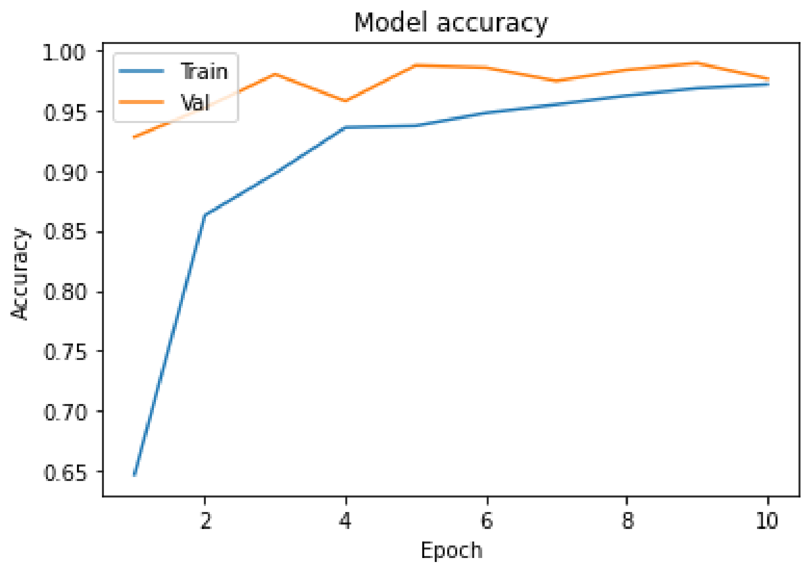

Figure 5-1 Training Model Accuracy

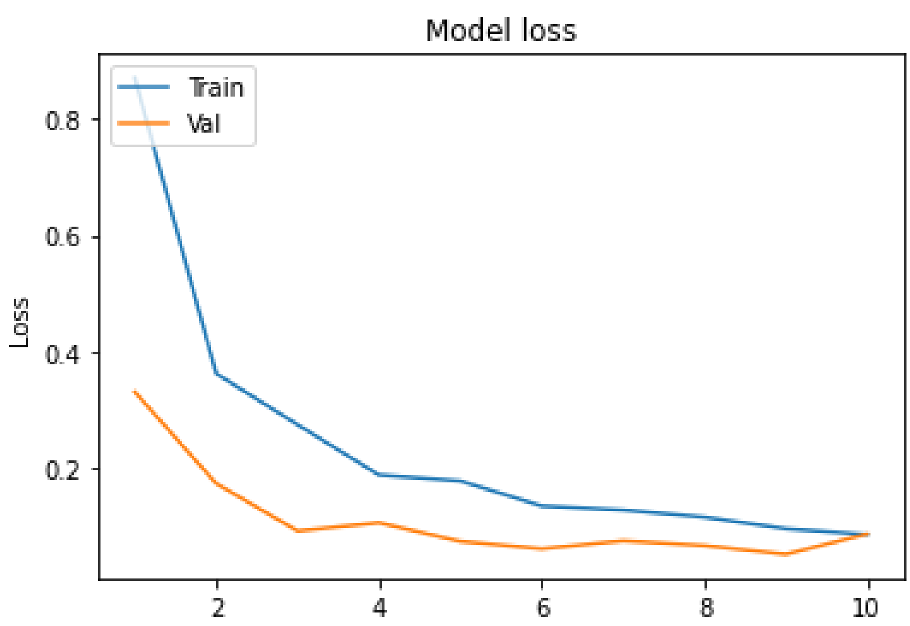

Figure 5-2 Training Model Loss

Similarly in following study [37], a brief work was presented about HAR. Details were given about different parameter increase can cause which type of effect in accuracy and training. We can witness from the confusion matric of following study that was a result of prediction in [37], that there was classification problem between three states of activities, sitting, standing and laying. For this study, the data acquisition was via Smart phone for Human activity recognition, while gathering data from smartphone may be the smart approach but because of artifacts and many other parameters the data obtained via smartphone is not much accurate as compared to dedicatedly designed system of wearable sensors for data collection. Thus, as compared to following study, we used dedicated system of wearable sensor for data acquisition. Not only this, but data was labelled and pre-processed properly. As shown in Figure 5-1, Figure 5-2, confusion matrix clearly classifying between four different activities. Still as per study in [37] 94.70% accuracy was achieved.

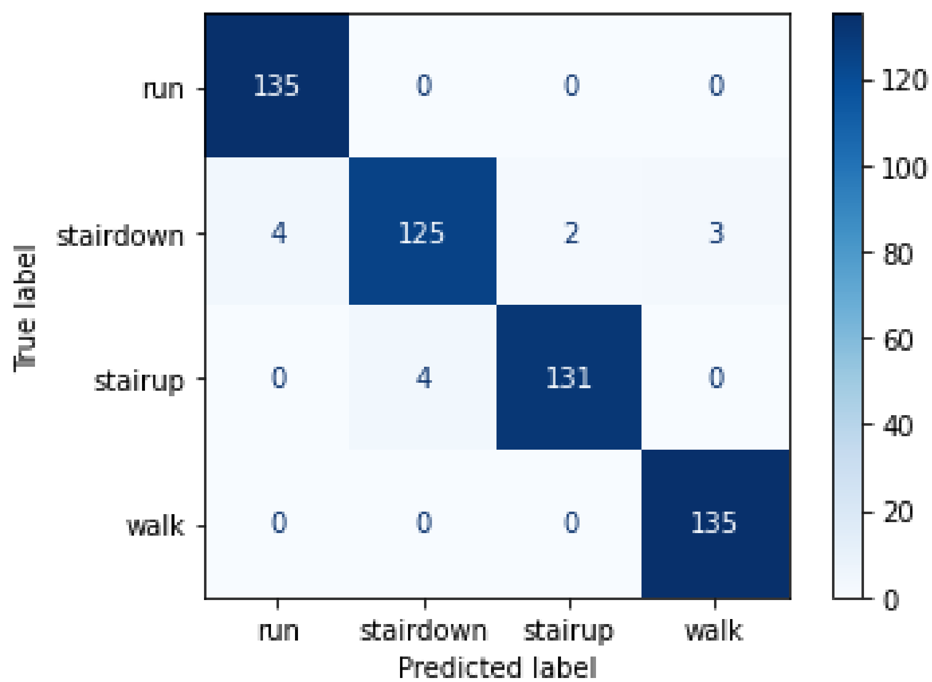

Figure 5-3 Training Confusion Matrix 1

In study [40], a method based on CNN was developed which was able to deal with scale invariance and local dependency of signal. The experiment was performed on three public data sets like Skoda, Opportunity, Anti tracker dataset and gave good accuracy w.r.t existing methods. Mainly there would three main tasks of this study, firstly, an approach is proposed based on CNN for feature extraction without any prior knowledge. Secondly, the proposed methods would be dealing with scale invariance and local dependencies. Thirdly, experimental results are given on these 3 public datasets. Following figure shows detailed structure of human activity recognition. Similarly, if results were observed according to confusion matrix shown in Fig1-24, we come to know that following study faced problem in classification of walking, walking up and walking down. As the reason from author was presented that because they are very closely related or similar, this was the cause of classification between walking, walking up and walking down. The main goal of following study was not to achieve accuracy but to introduce effect of max-pooling and other parameters in results via confusion matrix.

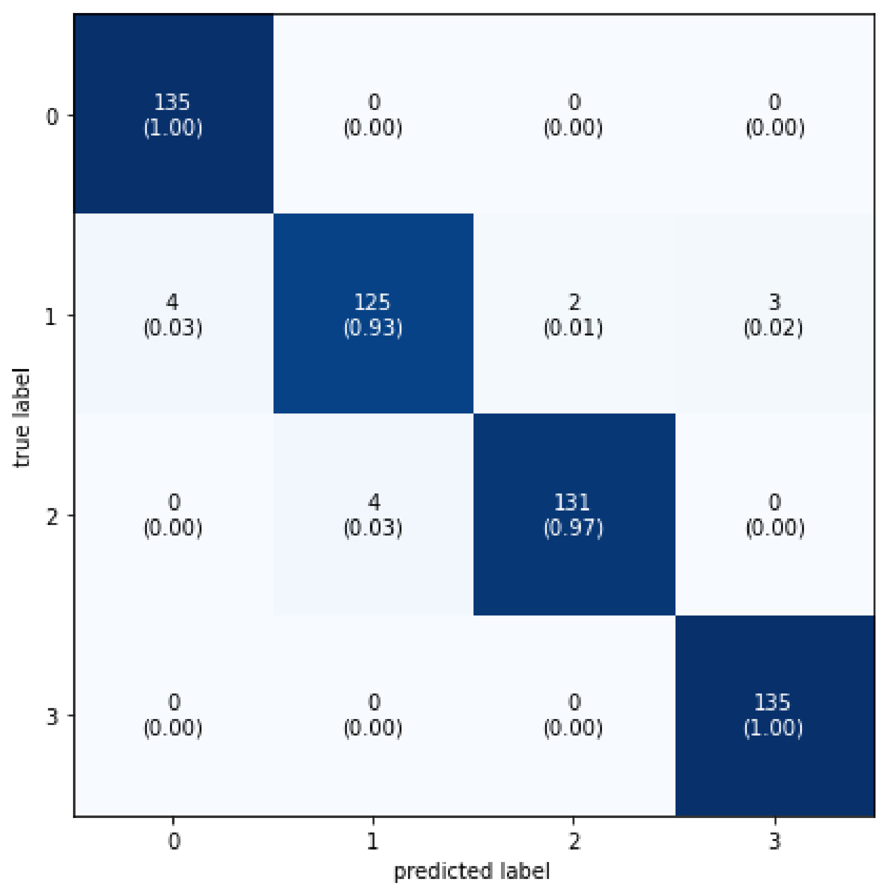

Figure 5-4 Training Confusion Matrix 2

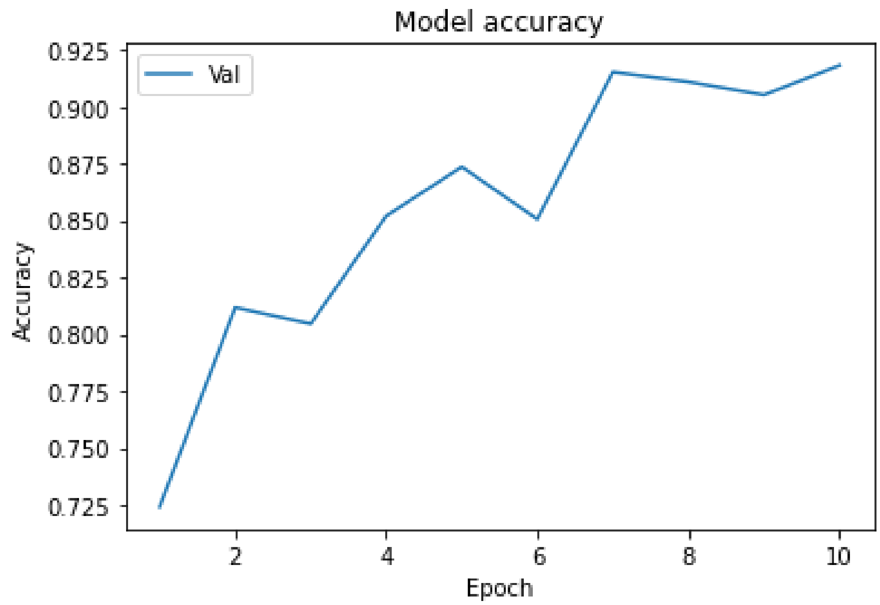

Figure 5-5 Real Time Model Accuracy

Similarly, following results were w.r.t to training and classification on computing platform used e.g., Google Collaboratory. Following results below were extracted after real time implementation of following algorithm on raspberry pi. On real time, form healthy person

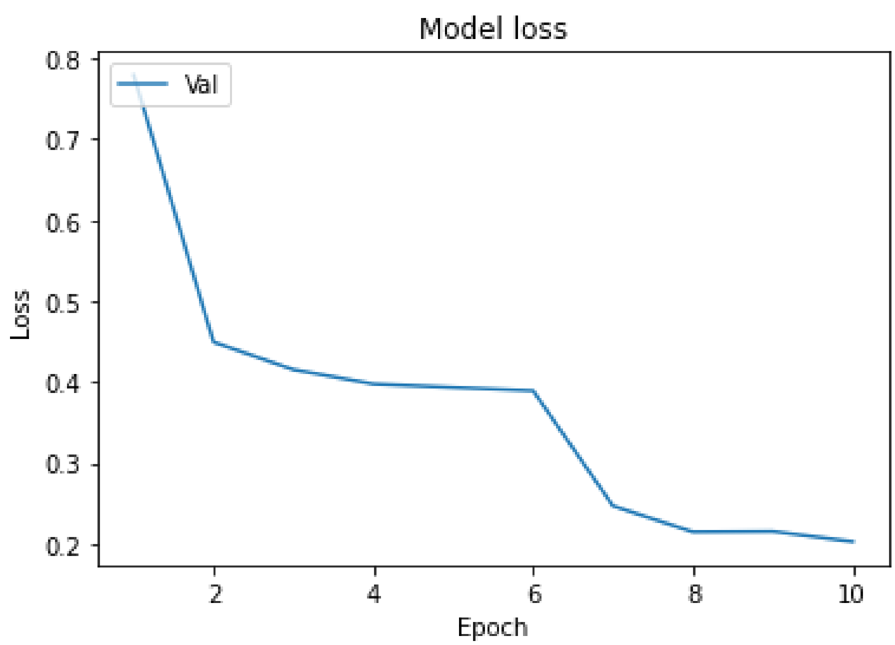

Figure 5-6 Real Time Model Loss

data is transmitted wirelessly using data acquisition hardware as discussed previously. Data is received at Raspberry pi. Figure 5-6,Figure 5-5 shows results.

Similarly, the following prediction times were also calculated so that w.r.t to time following HAR could be seen. In the figure, time relation is shown with prediction. One batch is selected as sample for studying relation between time and prediction. Approximately, it took 1.7-1.9 s to predict as shown in figure 5-8. But some part of graph shows values to 2.6-2.7s. This is basically caused due to following reasons

1) Hardware Processing Limitation
2) Signal distortion
3) Values Received has some issues.
4) Single Thread Coding rather than multi-threading coding.

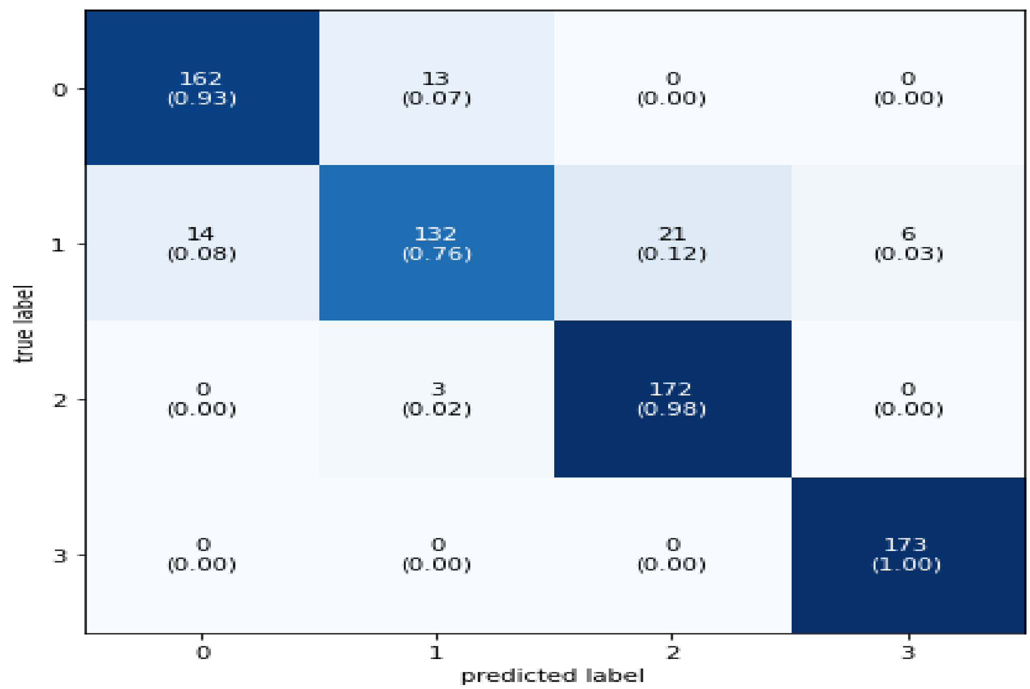

Figure 5-7 Real Time Confusion Matrix

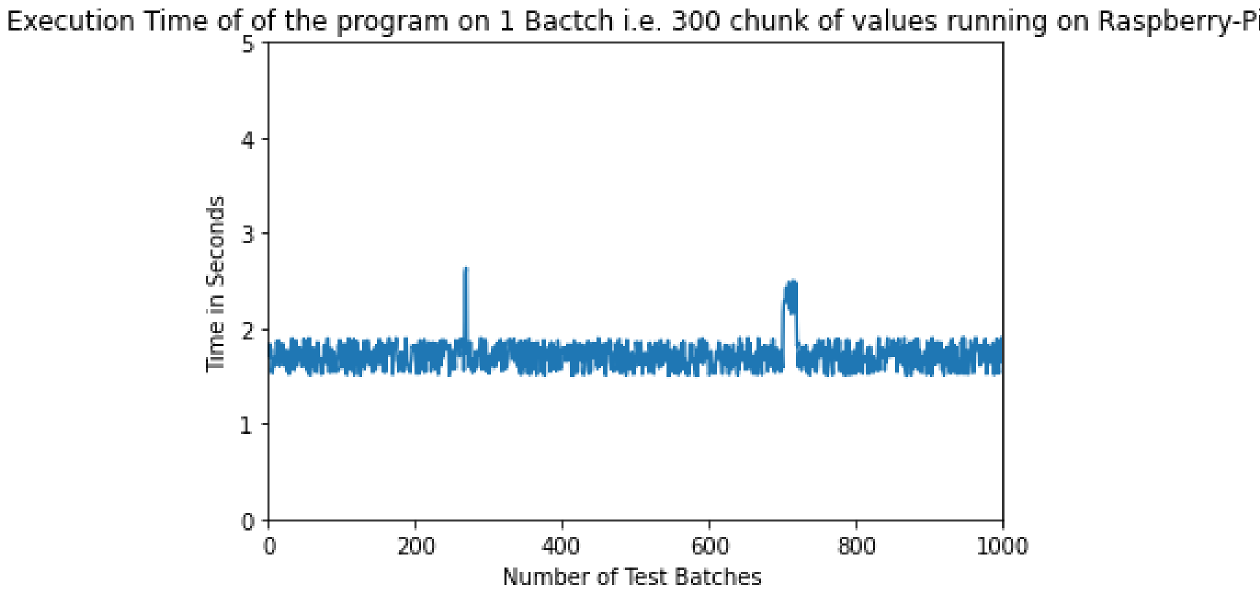

Figure 5-8 Prediction Time

5.2 Future Recommendations:

This Thesis, consist up of involvement of deep learning, hardware interface, controller interfaces etc. Following recommendations are suggested in regard to this thesis

1) Optimization of training and testing of algorithm by eliminating the parameters. This can be done via multiple iteration of experiment with different set of parameter or comparison of most effective parameter. Based on that algorithm should be trained. This would help to reduce computation time, prediction accuracy etc.
2) Use of multi-threading program then single threading. As currently at raspberry pi side only one program has been executed which is receiving data in serial. As there is a buffer which waits for 300 chunks of values. Once its completed prediction is executed. If there is some error while data has been stored in buffer, the whole chunk of values are rejected. Thus, this results in delay. So, if multi-thread programming is done in which one independent code which is receiving data serially and storing it. Another prediction code which is also working independently, keeps on predicting as receives values. In this there is very less chance of delays.

3) Increase of Hardware with advanced specification. As raspberry pi has 8gb ram with good processing but when it comes to real time prediction with accuracy, we need highly computing hardware to process things smoothly
4) Optimization of design of Prosthetic knee. As currently the location of valves of pneumatic are at such position that difficult coupling is made. Secondly, proper controlled closed feedback loop is required at knee hardware and states which are defined for respective activity.
5) Following work is performed in controlled environment for data set collection and prediction of results. If further following system is made robust and adjustable to every person gate cycle automatically. As right now gate cycle is not automatically adjusted as a standard gate cycle is taken for processing.

Bibliography

[1] Lee, S.M.; Yoon, S.M.; Cho, H Human activity recognition from accelerometer data using Convolutional Neural Network. In Proceedings of the 2017 IEEE International Conference on Big Data and Smart Computing (BigComp), Jeju, Korea, 13–16 February 2017; pp. 131–134

[2] Capela, N.; Lemaire, E.; Baddour, N.; Rudolf, M.; Goljar, N.; Burger, H. Evaluation of a smartphone human activity recognition application with able-bodied and stroke participants. J. Neuroeng. Rehabil. 2016, 13, 5.

[3] Wang, A.; Chen, G.; Yang, J.; Zhao, S.; Chang, C.Y. A comparative study on human activity recognition using inertial sensors in a smartphone. IEEE Sens. J. 2016, 16, 4566–4578

[4] O. D. Lara and M. A. Labrador, “A survey on human activity recognition using wearable sensors,” IEEE Communications Surveys Tutorials, vol. 15, no. 3, pp. 1192–1209, Third 2013.

[5] D. Cook, K.D. Feuz, N.C. Krishnan, Transfer learning for activity recognition: a survey, Knowl. Inf. Syst. 36 (3) (2013) 537–556

[6] Y. Bengio, Deep learning of representations: Looking forward, in: International Conference on Statistical Language and Speech Processing, Springer, 2013, pp. 1–37

[7] M.A. Alsheikh, A. Selim, D. Niyato, L. Doyle, S. Lin, H.-P. Tan, Deep activity recognition models with triaxial accelerometers, AAAI Workshop (2016).

[8] N.D. Lane, P. Georgiev, L. Qendro, Deepear: robust smartphone audio sensing in unconstrained acoustic environments using deep learning, in: UbiComp, ACM, 2015, pp. 283–294.

[9]N. W. Fonts. Nine Web Fonts. http://www.onlinewebfonts.com/icon, accessed March 15, 2017

[10] E. Fleishman, M. Quaintance, and L. Broedling. Taxonomies of human performance: the description of human tasks. Academic Press, 1984

[11] D. Ferguson. Therbligs: The Keys to Simplifying Work, 2000. http://gilbrethnetwork.tripod.com/therbligs.html, accessed April 21, 2017.

[12]O. D. Lara and M. a. Labrador. A Survey on Human Activity Recognition using Wearable Sensors. IEEE Communications Surveys & Tutorials, 15(3):1192–1209, 2013

[13]S. Dernbach, B. Das, N. C. Krishnan, B. Thomas, and D. Cook. Simple and Complex Activity Recognition through Smart Phones. In International Conf. on Intelligent Environments, pages 214–221, 2012.

[14]Wang, Y., Cang, S., & Yu, H. (2019). A survey on wearable sensor modality centred human activity recognition in health care. Expert Systems with Applications

[15] Cook, D., Feuz, K. D., & Krishnan, N. C. (2013). Transfer learning for activity recognition: A survey. Knowledge and information systems, 36(3), 537-556.

[16]R. Chavarriaga, H. Sagha, A. Calatroni, S.T. Digumarti, G. Tröster, J.d.R. Millán, D. Roggen, The opportunity challenge: a benchmark database for on–body sensor-based activity recognition, Pattern Recognit. Lett. 34 (15) (2013) 2033–2042

[17] Y. Chen, Y. Xue, A deep learning approach to human activity recognition based on single accelerometer, in: Systems, Man, and Cybernetics (SMC), 2015 IEEE International Conference on, IEEE, 2015, pp. 1488–1492.

[18] W. Jiang, Z. Yin, Human activity recognition using wearable sensors by deep convolutional neural networks, in: MM, ACM, 2015, pp. 1307–1310

[19] T. Plötz, N.Y. Hammerla, P. Olivier, Feature learning for activity recognition in ubiquitous computing, in: IJCAI, 22, 2011, p. 1729

[20] Y. Kim, B. Toomajian, Hand gesture recognition using micro-doppler signatures with convolutional neural network, IEEE Access (2016).

[21]N.D. Lane, P. Georgiev, Can deep learning revolutionize mobile sensing? in: Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications, ACM, 2015, pp. 117–122.

[22]Yu Gu, Lianghu Quan, and Fuji Ren. 2014. Wifi-assisted human activity recognition. In 2014 IEEE Asia Pacific Conference on Wireless and Mobile. IEEE, 60–65.

[23] Siamak Yousefi, Hirokazu Narui, Sankalp Dayal, Stefano Ermon, and Shahrokh Valaee. 2017. A survey on behavior recognition using wifi channel state information. IEEE Communications Magazine 55, 10 (2017), 98–104.

[24] Guanhua Wang, Yongpan Zou, Zimu Zhou, Kaishun Wu, and Lionel M Ni. 2016. We can hear you with wi-fi! IEEE Transactions on Mobile Computing 15, 11 (2016), 2907–2920.

[25] Kamran Ali, Alex X Liu, Wei Wang, and Muhammad Shahzad. 2015. Keystroke recognition using wifi signals. In Proceedings of the 21st Annual International Conference on Mobile Computing and Networking. ACM, 90–102

[26] R. Chavarriaga, H. Sagha, A. Calatroni, S.T. Digumarti, G. Tröster, J.d.R. Millán, D. Roggen, The opportunity challenge: a benchmark database for on– body sensor-based activity recognition, Pattern Recognit. Lett. 34 (15) (2013) 2033–2042.

[27] Xinyu Li, Yanyi Zhang, Mengzhu Li, Ivan Marsic, JaeWon Yang, and Randall S Burd. 2016. Deep neural network for RFID-based activity recognition. In Proceedings of the Eighth Wireless of the Students, by the Students, and for the Students Workshop, S3@MobiCom 2016. ACM, 24–26.

[28] Ki-Seung Lee. 2019. Joint Audio-ultrasound food recognition for noisy environments. IEEE journal of biomedical and health informatics (2019)

[29] Jingyuan Cheng, Mathias Sundholm, Bo Zhou, Marco Hirsch, and Paul Lukowicz. 2016. Smart-surface: Large scale textile pressure sensors arrays for activity recognition. Pervasive and Mobile Computing 30 (2016), 97–112

[30] Nicholas Foubert, Anita M McKee, Rafik A Goubran, and Frank Knoefel. 2012. Lying and sitting posture recognition and transition detection using a pressure sensor array. In 2012 IEEE International Symposium on Medical Measurements and Applications Proceedings. IEEE, 1–6.

[31] Edward S Sazonov, George Fulk, James Hill, Yves Schutz, and Raymond Browning. 2010. Monitoring of posture allocations and activities by a shoe-based wearable sensor. IEEE Transactions on Biomedical Engineering 58, 4 (2010), 983–990

[32] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and Jorge Luis Reyes-Ortiz. 2013. A public domain dataset for human activity recognition using smartphones.. In Esann

[33] Song-Mi Lee, Sang Min Yoon and Heeryon Cho, "Human activity recognition from accelerometer data using Convolutional Neural Network," 2017 IEEE International Conference on Big Data and Smart Computing (BigComp), 2017, pp. 131-134, doi: 10.1109/BIGCOMP.2017.7881728.

[34] G. Ascioglu and Y. Senol, "Design of a Wearable Wireless Multi-Sensor Monitoring System and Application for Activity Recognition Using Deep Learning," in IEEE Access, vol. 8, pp. 169183-169195, 2020, doi: 10.1109/ACCESS.2020.3024003.

[35] Sousa Lima, W., Souto, E., El-Khatib, K., Jalali, R., & Gama, J. (2019). Human activity recognition using inertial sensors in a smartphone: An overview. Sensors, 19(14), 3213.

[36] Ronao, C., & Cho, S. (2015). Deep Convolutional Neural Networks for Human Activity Recognition with Smartphone Sensors. Neural Information Processing, 46-53. https://doi.org/10.1007/978-3-319-26561-2_6

[37] B. Yang, M. N. Nguyen, P. P. San, X. L. Li, and S. Krishnaswamy,“Deep convolutional neural networks on multichannel time series for human activity recognition,” in Proceedings of the International Joint Conference

[38] Slim, S., Atia, A., M.A., M., & M.Mostafa, M. (2019). Survey on Human Activity Recognition based on Acceleration Data. International Journal Of Advanced Computer Science And Applications, 10(3). https://doi.org/10.14569/ijacsa.2019.0100311

[39] Son, M.; Jung, S.; Jung, S.; Hwang, E. BCGAN: A CGAN-based over-sampling model using the boundary class for data balancing. *J. Supercomput.* **2021**, *77*, 10463–10487

[40] M. Zeng et al., "Convolutional Neural Networks for human activity recognition using mobile sensors," 6th International Conference on Mobile Computing, Applications and Services, 2014, pp. 197-205, doi: 10.4108/icst.mobicase.2014.257786.

[41] A. Bulling, U. Blanke, B. Schiele, A tutorial on human activity recognition using body-worn inertial sensors, ACM Comput. Surv 46 (3) (2014) 33.

[42] Wang, J., Chen, Y., Hao, S., Peng, X., & Hu, L. (2019). Deep learning for sensor-based activity recognition: A survey. Pattern Recognition Letters, 119, 3-11. https://doi.org/10.1016/j.patrec.2018.02.010

[43] Chen, K., Zhang, D., Yao, L., Guo, B., Yu, Z., & Liu, Y. (2022). Deep Learning for Sensor-based Human Activity Recognition. ACM Computing Surveys, 54(4), 1-40. https://doi.org/10.1145/3447744

[44] Nweke, H. F., Teh, Y. W., Al-garadi, M. A., & Alo, U. R. (2018). Deep learning algorithms
for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges. Expert Systems with Applications, 105, 233–261. 10.1016/j.eswa.2018.03.056

[45] Capela, N. A., Lemaire, E. D., & Baddour, N. (2015). Feature selection for wearable smartphone-based human activity recognition with able bodied, elderly, and stroke patients. Plos One, 10, e0124414

[46] Charalampous, K., & Gasteratos, A. (2016). On-line deep learning method for action recognition. Pattern Analysis and Applications, 19, 337–354

[47] Chen, Y., & Xue, Y. (2015). A deep learning approach to human activity recognition based on single accelerometer. In Systems, man, and cybernetics (SMC), 2015 IEEE international conference on (pp. 1488–1492). IEEE

[48] Eskofier, B. M., Lee, S. I., Daneault, J.-F., Golabchi, F. N., Ferreira-Carvalho, G., Vergara-Diaz, G., et al. (2016). Recent machine learning advancements in sensor-based mobility analysis: Deep learning for Parkinson's disease assessment. In Engineering in Medicine and Biology Society (EMBC), 2016 IEEE 38th Annual International Conference of the (pp. 655–658). IEEE

[49] Gjoreski, M., Gjoreski, H., Luštrek, M., & Gams, M. (2016). How accurately can your wrist device recognize daily activities and detect falls? Sensors, 16, 800

[50] Gong, J.Q., Goldman, M.D., & Lach, J. (2016). DeepMotion: A deep convolutional neural network on inertial body sensors for gait assessment in multiple sclerosis. 2016 Ieee Wireless Health (Wh), 164–171.

[51] Kautz, T., Groh, B. H., Hannink, J., Jensen, U., Strubberg, H., & Eskofier, B. M. (2017). Activity recognition in beach volleyball using a Deep Convolutional Neural Network. Data Mining and Knowledge Discovery, 1–28

[52] Pouladzadeh, P., Kuhad, P., Peddi, S. V. B., Yassine, A., & Shirmohammadi, S. (2016). Food calorie measurement using deep learning neural network. In Instrumentation and measurement techn

[53] G. M. Weiss, "WISDM Smartphone and Smartwatch Activity and Biometrics Dataset," p. 5.

[54] Dataset Collection and Mechanism Design for Semi-Automation of Prosthetic Pneumatic Knee Joint,Hamza Qurashi,2022

[55] Raspberry Pi 4 8GB, MPU6050. [online] Available at: https://www.canakit.com/raspberry-pi-4-8gb.html> [Accessed 27 June 2022].