5-2022

# Improved Sensor-Based Human Activity Recognition Via Hybrid Convolutional and Recurrent Neural Networks

Sonia Perez-Gamboa

# IMPROVED SENSOR-BASED HUMAN ACTIVITY RECOGNITION VIA HYBRID CONVOLUTIONAL AND RECURRENT NEURAL NETWORKS

---

A Project

Presented to the

Faculty of

California State University,

San Bernardino

---

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in

Computer Science

---

by

Sonia Perez-Gamboa

May 2022

# IMPROVED SENSOR-BASED HUMAN ACTIVITY RECOGNITION VIA HYBRID CONVOLUTIONAL AND RECURRENT NEURAL NETWORKS

---

A Project

Presented to the

Faculty of

California State University,

San Bernardino

---

by

Sonia Perez-Gamboa

May 2022

Approved by:

Qingquan Sun, Advisor, Computer Science & Engineering

George Georgiou, Committee Member

Haiyan Qiao, Committee Member

## ABSTRACT

Non-intrusive sensor-based human activity recognition is utilized in a spectrum of applications including fitness tracking devices, gaming, health care monitoring, and smartphone applications. Deep learning models such as convolutional neural networks (CNNs) and long short-term memory (LSTMs) recurrent neural networks provide a way to achieve human activity recognition accurately and effectively. This project designed and explored a variety of multi-layer hybrid deep learning architectures which aimed to improve human activity recognition performance by integrating local features and was scale invariant with dependencies of activities. We achieved a 94.7% activity recognition rate on the University of California, Irvine public domain dataset for human activity recognition containing 6 activities with a 2-layer CNN-1-layer LSTM hybrid model. Additionally, we achieved an 88.0% activity recognition rate on the University of Texas at Dallas Multimodal Human Activity dataset containing 27 activities with a 4-layer CNN-1-layer LSTM hybrid model. For both datasets, our hybrid models outperformed other deep learning models and traditional machine learning methods.

ACKNOWLEDGEMENTS

I would like to acknowledge all who supported me the last few years. First, my advisor and committee members, each of whom have provided me guidance during my undergraduate and graduate journeys. Thank you for serving as mentors and pushing me to do more than I ever thought I was capable of.

Secondly, I would like to acknowledge my family and friends, who have immersed me in unwavering support and love and have gotten me where I am today.

Lastly, I would like to acknowledge every staff and faculty member at CSUSB who has helped me at any point during my time as a student. Thank you for all your help, resources, and for guiding me when I was lost.

# DEDICATION

I would like to dedicate this work to my nieces and nephews: Evelyn, Elijah, Josiah, Zoe, Mia, and Rhiannon. You are the greatest joys in my life. Thank you for making me the proudest, happiest *tía*. May this be a reminder that you can accomplish anything you set your mind to.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

# CHAPTER ONE

# INTRODUCTION

## Background

Human activity recognition (HAR) is the ability of a system to properly detect and identify specific human activities by analyzing data that is typically collected through a sensor or camera. HAR is utilized in a spectrum of applications such as fitness tracking devices, monitoring the care of elders [1], gaming [2], health care monitoring [3], and smart homes [4]. Fitness tracking devices such as smartwatches and activity tracking bands allow for non-intrusive, automated collection of user data that can be recorded and analyzed in companion applications to provide insight into the user's performance. These devices are able to automatically identify the activity the user is performing, removing the need for the user to manually track their activity, and allowing for more data analysis. In [1], researchers developed a small, compact system that can be worn by elderly people living alone. Their activity can be monitored remotely by their family or caregiver, and they can be alerted if the person falls down. In [2], a mobile game application was controlled by the movements and breathing of the user. HAR has also been used in health care monitoring, where a recovering patient's fine motor skills were monitored, and the therapy was adjusted accordingly [3]. HAR is used to observe the behavioral interaction between people, as was used in [4], where the activity was analyzed to determine if there was a conflict between people in a smart home setting.

The ever-growing demand for applications that can assist in not just these use cases, but across all domains, reinforces the need to determine the most efficient method of HAR. Several studies have adopted traditional machine learning methods for HAR, but these methods include the major drawback of requiring an expert in the field to complete necessary feature extraction before data can be classified. Contrary to previously used techniques, deep learning methods are capable of completing feature extraction without requiring a human expert. Deep learning utilizes artificial neural networks, with convolutional neural networks (CNNs) and recurrent neural networks (RNNs) being some of the most widely used for HAR.

## Purpose

This research aims to develop a hybrid deep learning model that utilizes both CNNs and RNNs, specifically long short-term memory (LSTM) RNNs, to increase the overall recognition rate when applied to sensor-based HAR. The motivation behind this research is to 1) support the idea that deep learning methods yield high accuracy of HAR when compared to traditional machine learning, and 2) improve the performance of deep learning models by presenting a lightweight, hybrid, multi-layer deep learning model that achieves a balance between high recognition rate and training time consumption. The use of HAR is a task used across many domains whose purpose is to automate the recognition of simple or specialized activities. Therefore, it is important to find the most efficient method to accomplish this.

# CHAPTER TWO

# HUMAN ACTIVITY DATA

## Collection Methods

Choosing the best modality for recording human activity is the first step in accomplishing HAR. Some of the most common systems to use include optical motion capture systems, simple cameras, and wearable sensors. Optical motion capture systems utilize infrared cameras and reflective sensors. Reflective sensors are placed on a subject at major joints and areas of interest. The cameras then emit infrared light and capture the reflection off the sensors. Optical motion capture systems provide very accurate human activity data but are considerably more expensive than other modalities. Image and videos recorded through simple cameras provide us with accurate mediums and are at the core of current computer vision research. Some obstacles presented by optical motion capture systems and simple cameras are: 1) Images and videos can capture surrounding movements that are not part of the human activity, 2) the captures can be negatively affected by lighting or other elements in the environment, and 3) since body parts can be blocked by other body parts or camera angles, it is necessary to increase the number of cameras in both of these systems to increase the accuracy, leading to an increase in cost. Wearable sensors such as gyroscopes, accelerometers, heart monitors, and electrodes are small sensors that do not have the limitations of cameras and optical motion captures systems and still provide accurate recordings of human activity.

Additionally, they are inexpensive, have low energy consumption, are small, and, therefore, are non-intrusive.

## Inertial Sensors

Our research focused on exploring the use of inertial sensors to collect human activity data. Inertial sensors are sensors that record specific gravity and angular rates of the subjects or objects to which they are attached to. Inertial sensors consist of gyroscopes, accelerometers, and an optional magnetometer. Accelerometers provide measurements of linear acceleration on 3 axes, while gyroscopes provide measurements of angular velocity on 3 axes. Inertial sensors can vary in size, but our research focused on small, non-intrusive, wearable inertial sensors.

The embedded gyroscope and accelerometer of a Samsung Galaxy S II smartphone were used to collect inertial measurements on subjects in [5]. The smartphone measures 4.93 inches (H) x 2.6 inches (W) x 0.334 inches (D), and weighs about 4.1 ounces, making it lightweight and easy for subjects to wear. Researchers in [6] used a micro electro-mechanical systems (MEMS) sensor to capture acceleration and angular velocity data. The sensor is similar in size to a U.S quarter, which measures about 0.945 inches in diameter, making it small and non-intrusive. Figure 1 shows the smartphone used in [5], and Figure 2 shows the inertial sensor used in [6]. Subjects in both studies wore the sensors on single locations of their body and completed several activities multiple times. The

sensors captured the acceleration and angular velocity measurements, and researchers were then able to extract the raw inertial sensor signals.

Figure 1. Samsung Galaxy S II Phone and Wearable Case.

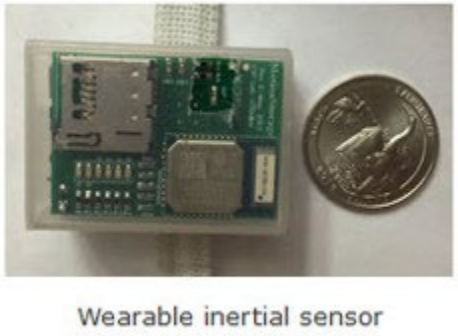


Figure 2. MEMS Inertial Sensor.

# CHAPTER THREE

# HUMAN ACTIVITY RECOGNITION

## Traditional Machine Learning vs. Deep Learning

Once the human activity data containing raw sensor signals are obtained, the next step in the HAR process is to choose the best method to properly analyze the data. For years, artificial intelligence has been used to accomplish recognition and classification problems. Artificial Intelligence (AI) is the concept of constructing computers, or machines, in such a way that they possess the same characteristics as human intelligence. One way to accomplish AI is through traditional machine learning (TML), which uses different algorithms to analyze data, learn from it, and then make a prediction or classification about something related to the data. The idea is to introduce sufficient data to a machine so it may learn enough from it to properly predict or classify a new piece of information. Some common TML algorithms used for sensor-based HAR are support vector machines (SVM), collaborative representation classifiers (CRC), decision trees, discriminant analysis, nearest neighbor classifiers, and ensemble classifiers. In [5], [7], and [8], Multi-Class SVMs (MC-SVM) and Multiclass Hardware Friendly SVMs (MC-HF-SVM) were used to successfully classify several simple human activities. In [9], researchers compared the performance of over 20 different TML algorithms on 5 simple activities.

As successful as TML methods have been in sensor-based HAR, the process is not completely automated. A critical step in the classification process

for TML, described in Figure 3, is the necessity of a human expert within the domain to manually extract features that the TML algorithm needs to make predictions. This requirement for feature extraction limits the flexibility of these methods.

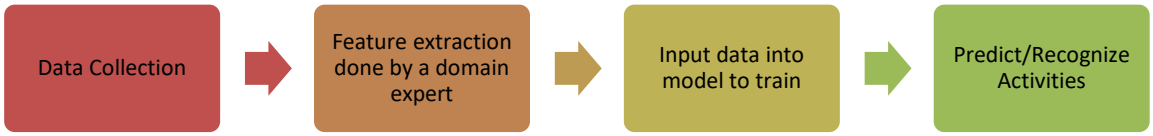


Figure 3. Process for HAR with Traditional Machine Learning.

Another pitfall of TML is its performance as the amount of input data is increased. With advances in technology and accessibility to very large amounts of data, the goal is for algorithm performance to proportionally increase with the amount of data available. Unfortunately, research has shown that as the amount of input data for TML algorithms increases, the performance of the algorithms plateaus [10]. This lack of improvement means that TML cannot fully take advantage the large amounts of data available. The drawbacks of TML bring attention to a different subset of AI, which introduces a more efficient approach to the HAR problem: deep learning.

Deep learning is a method of machine learning that utilizes artificial neural networks to accomplish the tasks of automatic activity recognition and classification with little to no human intervention. Figure 4 presents the process of activity recognition with deep learning algorithms, showing there is no need for a human expert to complete feature extraction.

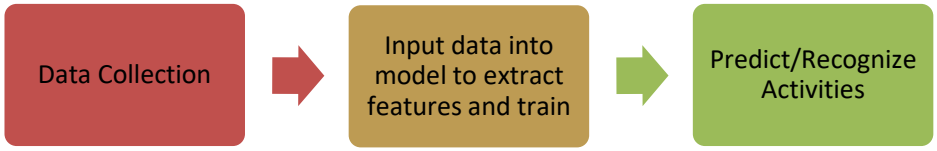


Figure 4. Process for HAR with Deep Learning.

Additionally, deep learning algorithms have been shown to increase in performance as the amount of data presented increases (see Figure 5).

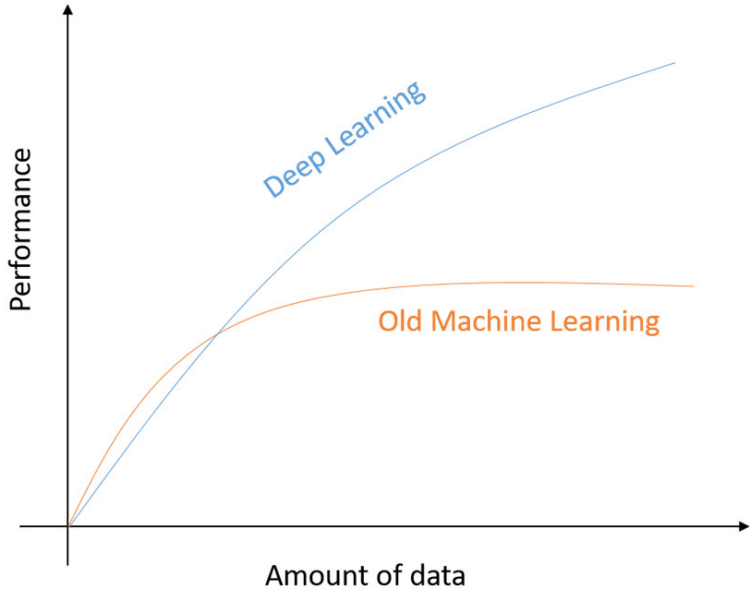


Figure 5. Performance of Algorithms vs. Amount of Data.
Alom, Md. Zahangir & Taha, Tarek & Yakopcic, Chris & Westberg, Stefan & Sidike, Paheding & Nasrin, Mst & Hasan, Mahmudul & Essen, Brian & Awwal, Abdul & Asari, Vijayan, "A State-of-the-Art Survey on Deep Learning Theory and Architectures," Electronics 8, no. 3:292, 2019.
https://doi.org/10.3390/electronics8030292

Deep learning models such as RNNs and CNNs have more recently been used to complete HAR due to their capability of automatically completing feature extraction on raw data without requiring a human expert, while also obtaining high recognition accuracy. Training deep neural networks can be computationally expensive, taking hours or several days to train models [11]. This project aims to compare the performance and accuracy of different deep learning models on sensor-based human activity datasets while achieving balance between recognition rate and total training time.

## Related Works

Deep learning techniques have been used across the field of machine learning to accomplish sensor-based HAR in different domains. CNNs use convolution to convolve over input signals and efficiently identify local patterns and create feature maps. CNNs have proven to work for HAR due to their capability of capturing local dependencies on signal data, as well as their preservation of feature scale invariance when completing feature extraction. In [12], [13], and [14], researchers present CNN models that achieved strong HAR accuracy when compared to other state-of-the-art approaches. In [12], they explore the effect that different parameter values have on the overall accuracy. It was found that pooling size, weight decay, and drop out must be modified relative to the number of different activities a dataset has, as well as the number of available samples in order to achieve the best accuracy.

LSTM RNNs have also been used to achieve HAR due to their ability to properly handle the long-term dependencies in time-series data such as sensor signals. In [15], researchers designed a multi-layer LSTM RNN model, which had a lower recognition time than CNN-based models it was compared to and had a higher recognition accuracy. In [16], a bidirectional LSTM outperformed regular CNN and regular LSTM models when applied to a large dataset. Researchers also found that RNNs such as LSTMs outperform CNNs in recognizing activities that are short in duration but have a natural ordering, such as opening and closing doors.

Recently, the combination of CNN and RNN models has been explored to further improve the performance of sensor-based HAR. A deep convolutional and recurrent model referred to as “DeepConvLSTM” was presented in [17], which achieved a higher HAR rate than a baseline CNN model. In [18], a combination of CNN, LSTM, and hybrid models were implemented to achieve HAR, with a 3-layer LSTM outperforming the other presented models. [19] compares the performance of baseline LSTM and CNN models against a hybrid model and found the hybrid to outperform the baseline models.

In [12]-[19], sensor-based human activity datasets of varying sizes are used to test all the implemented models. The studies include small datasets of 6-12 activities, medium datasets of 18 activities, and large datasets that include up to 46 gestures. Although the large datasets include many actions, they are very simple gestures that are used within specific work environments, such as assembly line workers. We wish to utilize datasets that include several complicated, highly correlated human activities.

Our work is based on a hybrid multi-layered CNN and LSTM model that presents the following contributions to the field of sensor-based HAR: 1) We design and implement a lightweight, multi-layer hybrid model that has high-performance accuracy when applied to simple and highly correlated activities; 2) we develop several CNN, LSTM, and hybrid models in the same environment to properly compare performances; and 3) we develop models that have a balance between high HAR accuracy and low training time.

# CHAPTER FOUR

# METHODOLOGY

To properly compare the performances of deep learning models on sensor-based human activity datasets, we chose to explore and implement several deep learning models including vanilla CNNs and LSTMs, multi-layer CNNs and LSTMs, and finally, hybrid multilayer CNN-LSTMs.

## Convolutional Neural Networks (CNN)

CNNs are deep learning models that utilize convolutional layers, pooling layers, fully connected layers, and hidden layers to accomplish classification and recognition. CNNs are popular in the field of computer vision, which performs classification and recognition of images and videos. Because of this, 2-Dimensional (2D) CNNs are used to properly handle the image and video inputs. Since our datasets contain raw signal data, we utilize 1-Dimensional (1D) CNNs, which are advantageous and preferable over 2D CNNs whenever possible because of their reduced complexity. Input signal data is fed into the convolutional layers, which convolve over the sequence. A convolution is a linear operation that multiplies an array of input data and a specified filter, with the filter being smaller than the input. The specific multiplication applied is the dot product, which multiplies the smaller filter and a filter-sized portion of the input and then sums the products. Since the filter is smaller than the input, this means it can be repeatedly applied across multiple sections of the input data until the whole input data is covered. The convolution process can be visualized with the following

equations: Given an input $x$, which is of length $m$; a filter $w$, which is of length $n$; the resulting sequence of dot products $y$ will be the same length as $x$ [20]

$$x = [x_0, x_1, x_2, ..., x_{m-1}]$$

$$w = [w_0, w_1, w_2, ..., w_{n-1}]$$

$$y = [y_0, y_1, y_2, ..., y_{m-1}]$$

It is common practice when implementing CNNs to have an odd filter size. Therefore, $n$ would have the following constraints: 1) $n < m$; 2) $n$ is odd; and 3) we can express the length of our filter as $n = 2p + 1$, where $p < \frac{n}{2}$ . We can update our notation to be

$$x = [x_0, x_1, x_2, ..., x_{m-1}]$$

$$w = [w_{-p}, w_{-p+1}, ..., w_0, ..., w_{p-1}, w_p]$$

$$y = [y_0, y_1, y_2, ..., y_{m-1}]$$

Considering the steps of the dot product, we can further expand the result $y$ as

$$y_0 = x_0 w_0 + x_1 w_{-1} + \cdots + x_p w_{-p}$$

$$y_1 = x_0 w_1 + x_1 w_0 + x_1 w_{-1} + \cdots + x_{p+1} w_{-p}$$

$$y_2 = x_0 w_2 + x_1 w_1 + \cdots + x_{p+2} w_{-p}$$

…

$$y_{m-1} = x_0 w_{m-1} + x_1 w_{m-2} + \cdots + x_{p+m-1} w_p$$

We can now use the summation shorthand to represent these equations more concisely:

$$y_j = \sum_{k=-p}^{p} x_{j-k} w_k$$

These convolutions identify local correlations within the input data and result in feature maps, which hold the exact location of detected features from the input data. The feature maps are then passed through a pooling layer, which reduces the sensitivity of the output feature map by down-sampling the detected features. Pooling helps the network identify the same feature, even if the exact location of the feature changes from one input sequence to the next [11]. The resultant feature maps are then fed into fully connected layers which combine different learned local structures and complete the final classification. CNNs have proven to effectively perform independent, non-handcrafted feature extraction on raw sensor data, which enhances the overall classification accuracy of the model [21].

For our sensor-based HAR case, we designed CNNs, whose architecture can be seen in Figure 6, with 1D convolutional layers, pooling layers, and dense layers. The chosen number of filters, kernel size, and the activation function were influenced by [12], although we made further modifications through trial and error as results were obtained.

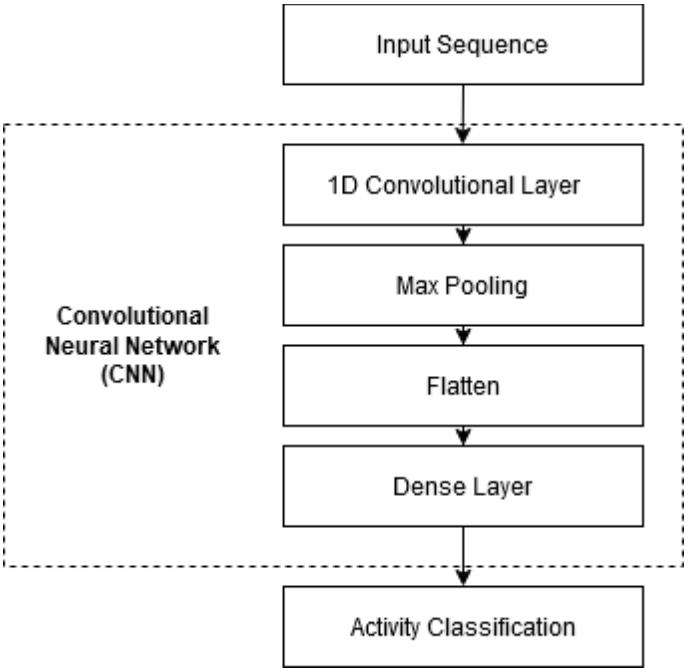


Figure 6. Baseline CNN Architecture.

## Long Short-Term Memory Neural Networks (LSTM)

Traditional feed-forward neural networks are models that are made up of input, hidden, and output layers where data moves in a forward motion without looping or going backward. Given a time series prediction problem such as an inertial signal, the value of a current time sample is influenced by previous time samples. Therefore, it is important for a network to take into consideration data that has already passed, making feed-forward networks unfit to handle time series data.

RNNs are special neural networks that can properly handle time series data or input sequences by feeding themselves information from past data to influence current data. For the current input sequence to be properly influenced, RNNs utilize back propagation through time (BPTT), which is a method of adjusting the weights that affect the training of a neural network by calculating the weight values that would result in the lowest loss. BPTT makes RNNs susceptible to exploding or vanishing gradients due to the constant derivation that occurs. Researchers developed an LSTM RNN to address this common issue with the addition of a special memory cell within each LSTM unit [22].

Hidden states in RNNs, including LSTMs, are variables that contain sequence information up to the current time step, $t$, meaning that the hidden state, $h_t$ , at any time step is influenced by the current input, $x_t$ , and previous hidden state, $h_{t-1}$ . The special memory cell in LSTMs is similar to hidden states, but its value is also influenced by additional gates. LSTMs utilize an output gate, input gate, and forget gate to determine what information is important enough to remember and what can be forgotten. Given $h$ hidden units, a batch size of $n$, and an input sequence of size $d$, the input is $X_t \in \mathbb{R}^{n \times d}$ and $h_{t-1} \in \mathbb{R}^{n \times h}$. Given this, the LSTM gates are defined as follows.

The input gate, $I_t$, which decides when data will be read into the memory cell, is calculated as

$$I_t = \sigma(X_t W_{xi} + h_{t-1} W_{hi} + b_i) ;$$

the forget gate, $F_t$, which decides what information can be removed from the cell, is calculated as

$$F_t = \sigma\left(X_t W_{xf} + h_{t-1} W_{hf} + b_f\right);$$

and lastly, the output gate, $O_t$, which reads out entries from the memory cell, is calculated as

$$O_t = \sigma(X_t W_{xo} + h_{t-1} W_{ho} + b_o);$$

where $W_{xi}$, $W_{xf}$, $W_{xo}$ , $W_{hi}$, $W_{hf}$, and $W_{ho}$ are weight parameters, $b_i$, $b_f$, and $b_o$ are bias parameters, and $\sigma$ is a sigmoid activation function.

Each of these gates influences the final value of the memory cell, which then enables the LSTM to learn and retain dependencies on long input sequences, which has been shown to work well for HAR using sensor data [23]. As the network processes more time steps, the memory cell "learns" based on the current input and past inputs, enabling it to properly retain information on hundreds of future inputs.

For our baseline LSTM model we combine LSTM and dense layers with 100 units each (see Figure 7).

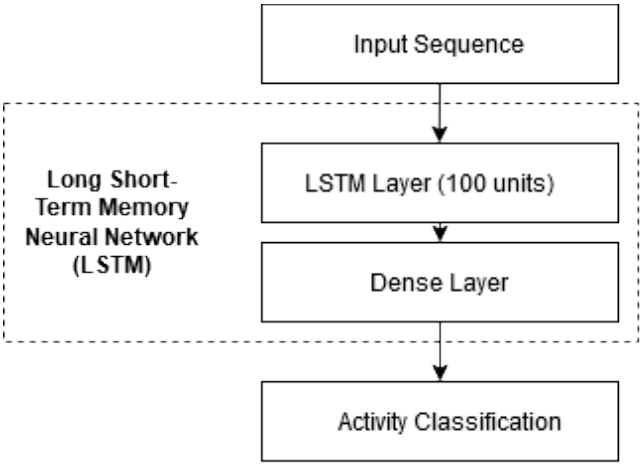


Figure 7. Baseline LSTM Architecture.

## CNN-LSTM Hybrid

After reviewing our baseline CNN and LSTM architectures, we designed a hybrid CNN-LSTM architecture that takes advantage of the feature extraction capabilities of a CNN and the retention of temporal dependencies of an LSTM. The architecture that we use is shown in Figure 8. We made modifications to this architecture by varying the number of CNN and LSTM layers to see how accuracy performance is affected. The process of HAR for the hybrid models is as follows: 1) Sensor data is input through 1D convolutional layer(s) which results in independent, non-handcrafted feature maps, 2) the output is passed through a max-pooling layer to down-sample the feature maps, 3) the remaining feature maps are then flattened to be processed through the LSTM layer(s), which identify temporal dependencies, and 4) the LSTM layer(s) output a vector of predictions which is passed through a softmax dense layer to complete the final

classification. We used this architecture for implementing a variety of hybrid architectures.

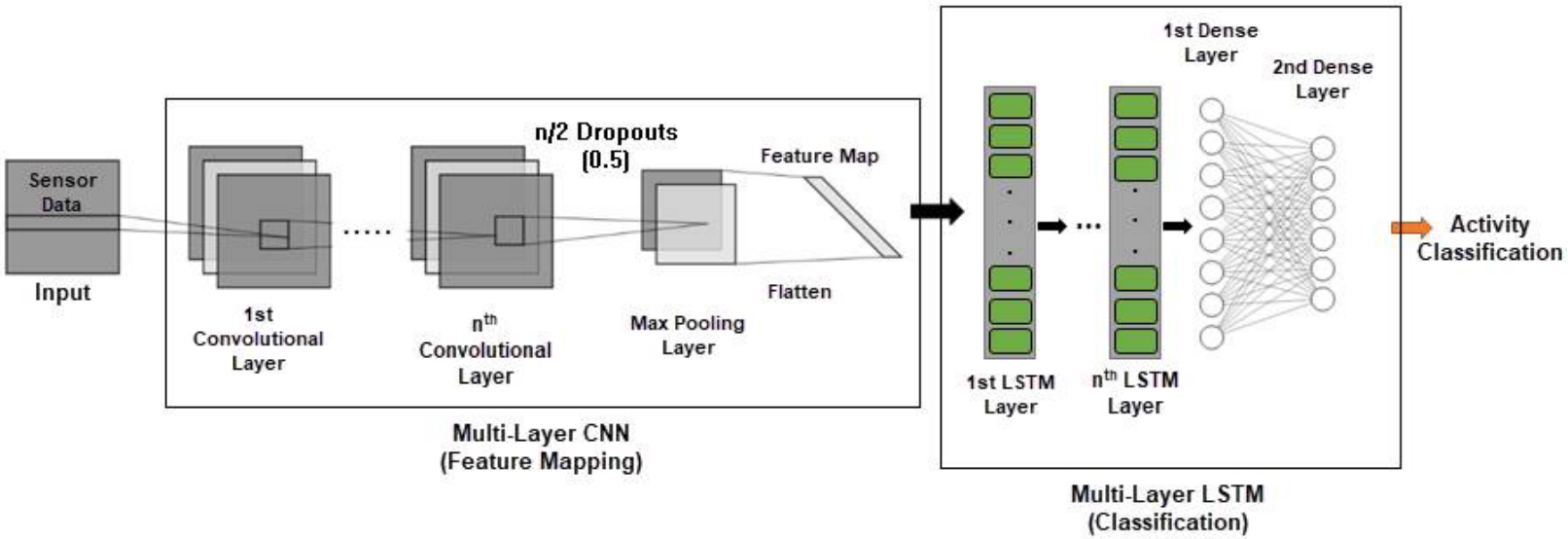


Figure 8: CNN-LSTM Hybrid Model
S. Perez-Gamboa, Q. Sun and Y. Zhang, "Improved Sensor Based Human Activity Recognition via Hybrid Convolutional and Recurrent Neural Networks," *2021 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL)*, 2021, pp. 1-4, doi: 10.1109/INERTIAL51137.2021.9430460.

# CHAPTER FIVE

# EXPERIMENT AND IMPLEMENTATION

## Datasets

I trained and tested our implemented models with the University of California, Irvine (UCI) public domain dataset for HAR [5] and the University of Texas at Dallas Multimodal Human Activity (UTD-MHA) dataset [6].

### UC Irvine Dataset

The first dataset we used to test our models is publicly available on the UCI Machine Learning repository. It is a dataset containing inertial data from the embedded accelerometer and gyroscope in a Samsung Galaxy S II smartphone (see Figure 1). A total of 30 subjects, ages ranging from 19 to 48, wore the smartphone on their waist and performed six daily living activities, twice: “standing”, “sitting”, “laying down”, “walking”, “walking downstairs”, and “walking upstairs”. Researchers collected the triaxial linear acceleration and angular velocity data at a sampling rate of 50Hz. Once all data was obtained, it was pre-processed using a median and 3$^{rd}$ order Butterworth filter. It was then fitted into 2.56-second fixed-width sliding windows with 50% overlap. Figures 9 and 10 provide images and graphs of the “walking” and “walking upstairs” actions and their corresponding acceleration data.

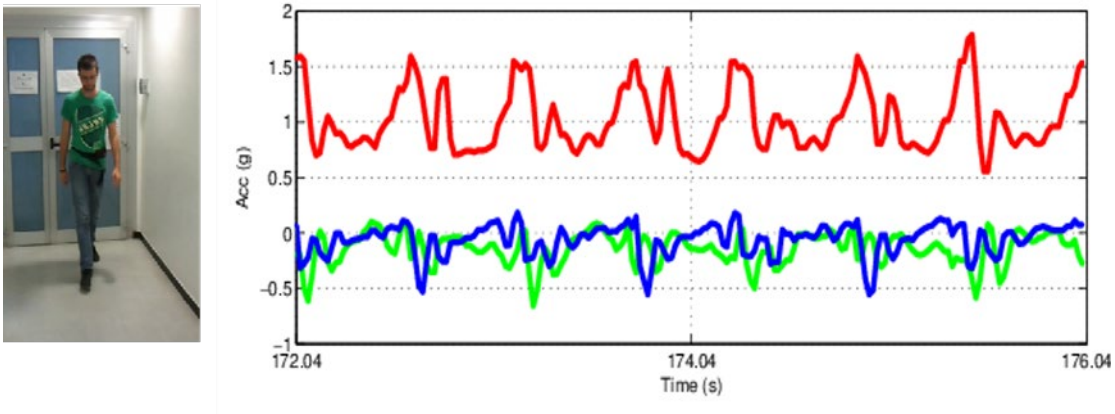


Figure 9. Image and Acceleration Signal for “Walking”.
D. Anguita, A. Ghio, L. Oneto, X.Parra and J. Reyes-Ortiz. “A Public Domain Dataset for Human Activity Recognition Using Smartphones,” 21th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2013. Bruges, Belgium 24-26 April 2013.

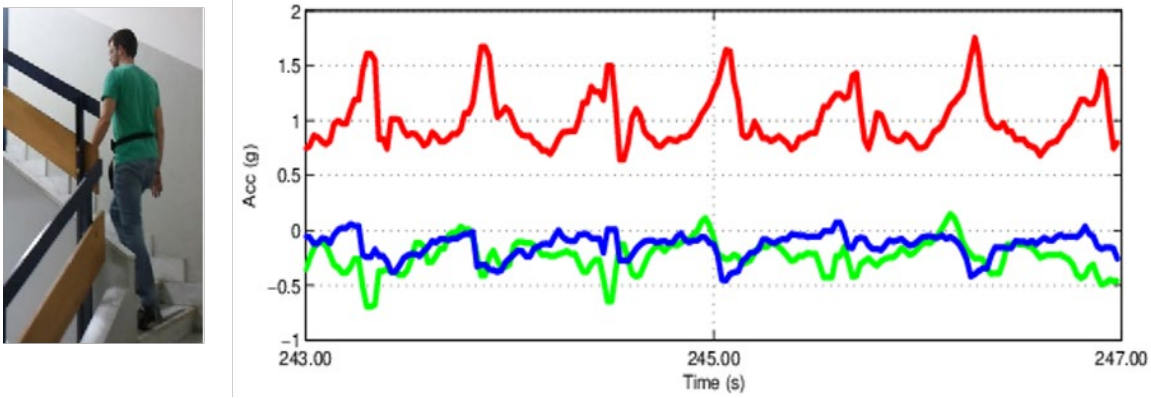


Figure 10. Image and Acceleration Signal for “Walking Upstairs”.
D. Anguita, A. Ghio, L. Oneto, X.Parra and J. Reyes-Ortiz. “A Public Domain Dataset for Human Activity Recognition Using Smartphones,” 21th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2013. Bruges, Belgium 24-26 April 2013.

UTD-MHA Dataset

The UTD-MHA dataset contains inertial sensor data that provides linear acceleration and angular velocity signals obtained from a low-cost wireless wearable inertial sensor that was built at the university (see Figure 2). Researchers in [6] had 8 subjects wear the inertial sensor and perform 27 different actions, 4 times. The subjects wore the inertial sensor on their right wrist for actions 1 through 21, and on their right thigh for actions 22-27. Table 1 has a full list of all 27 activities.

Table 1. Human Actions in UTD-MHA Dataset

| | | |
|---|---|---|
| Wearable inertial sensor on right wrist | | |
| 1 | *right arm swipe to the left* | *(swipe_left)* |
| 2 | *right arm swipe to the right* | *(swipe_right)* |
| 3 | *right hand wave* | *(wave)* |
| 4 | *two hand front clap* | *(clap)* |
| 5 | *right arm throw* | *(throw)* |
| 6 | *cross arms in the chest* | *(arm_cross)* |
| 7 | *basketball shoot* | *(basketball_shoot)* |
| 8 | *right hand draw x* | *(draw_x)* |
| 9 | *right hand draw circle (clockwise)* | *(draw_circle_CW)* |
| 10 | *right hand draw circle (counter clockwise)* | *(draw_circle_CCW)* |
| 11 | *draw triangle* | *(draw_triangle)* |
| 12 | *bowling (right hand)* | *(bowling)* |
| 13 | *front boxing* | *(boxing)* |
| 14 | *baseball swing from right* | *(baseball_swing)* |
| 15 | *tennis right hand forehand swing* | *(tennis_swing)* |
| 16 | *arm curl (two arms)* | *(arm_curl)* |
| 17 | *tennis serve* | *(tennis_serve)* |
| 18 | *two hand push* | *(push)* |
| 19 | *right hand knock on door* | *(knock)* |
| 20 | *right hand catch an object* | *(catch)* |
| 21 | *right hand pick up and throw* | *(pickup_throw)* |
| Wearable inertial sensor on right thigh | | |
| 22 | *jogging in place* | *(jog)* |
| 23 | *walking in place* | *(walk)* |
| 24 | *sit to stand* | *(sit2stand)* |
| 25 | *stand to sit* | *(stand2sit)* |
| 26 | *forward lunge (left foot forward)* | *(lunge)* |
| 27 | *squat (two arms stretch out)* | *(squat)* |

C. Chen, R. Jafari, and N. Kehtarnavaz, "UTD-MHAD: A Multimodal Dataset for Human Action Recognition Utilizing a Depth Camera and a Wearable Inertial Sensor"*, Proceedings of IEEE International Conference on Image Processing,* Canada, September 2015.

Researchers collected the signals at a sampling rate of 50Hz. Unlike the UCI dataset, which is provided pre-processed, these inertial signals were not pre-processed. It is common in signal processing to apply noise filters to remove non-vital information (“noise”). I applied a median filter, which is a non-linear filtering technique used to remove noise from images and signals by removing outlier data. I also applied a low pass 3$^{rd}$ order Butterworth filter with a cutoff frequency of 20Hz. Figure 11 shows an image of the “bowling” action and its

corresponding acceleration signal before and after filtering. The goal of noise filtering the data is to remove unnecessary noise without altering the overall signal to the point where it is no longer valid.

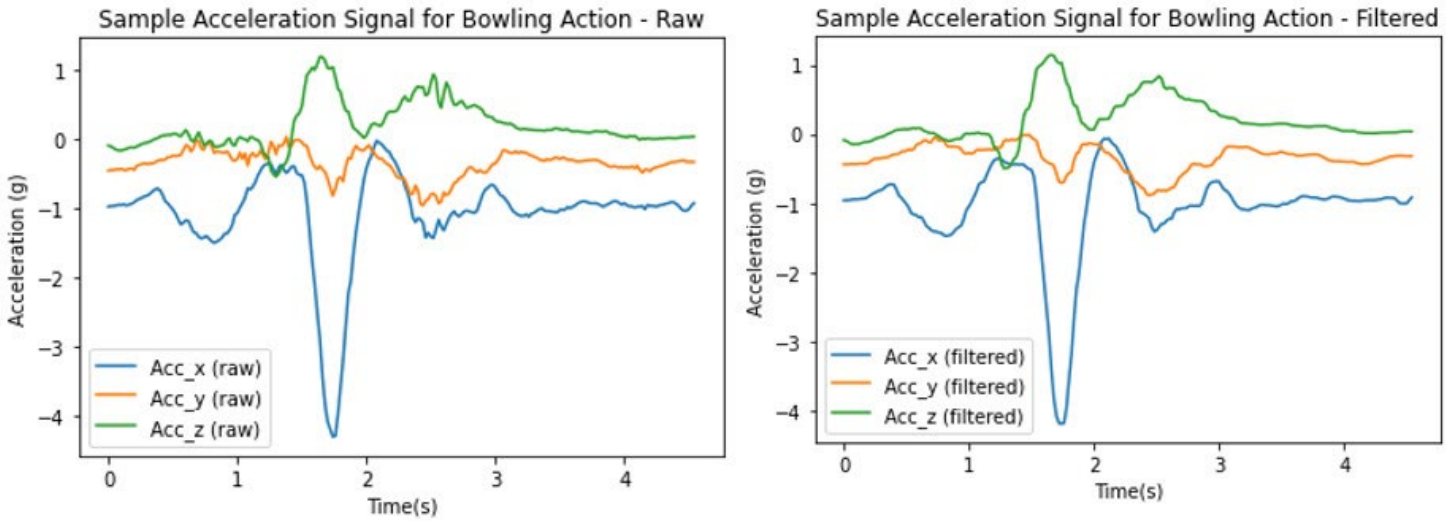


Figure 11. Image of “Bowling” Action (Top), Raw and Filtered Acceleration Signals (Bottom).

In addition to applying noise filters, it is also necessary to segment our data before feeding it into our deep learning models. The sliding window segmentation approach splits a raw sensor signal into windows of fixed size or fixed number of samples. Literature has shown that overlapping sliding windows increases the recognition rate in HAR [24]. Therefore, we segmented our data into windows of 144 time-steps with 50% overlap.

## Experiment Setup

For deep learning models, the most critical part of the HAR process is properly training the model on sufficient data so that it can accurately identify new, unseen data. I separated both the UCI and UTD-MHA datasets into 70% training and 30% testing subsets, with no overlap between the subsets.

To properly compare the performances of different deep learning models on the datasets, I designed the following: baseline CNN and LSTM models, multilayer CNN and LSTM models, and a variety of multilayer CNN-LSTM hybrid models. The number of epochs for training and hyperparameters such as the number of filters, kernels, dropout rates, and the number of nodes were selected by using [12] as a starting point and manually fine-tuning them through trial and error.

For the larger UTD-MHA dataset, I first ran all models with 15 activities, and then for all 27 activities. Starting with a lower total of activities allowed me to fine-tune the final models for the more complicated 27 activities.

## Implementation

Implementation of the models was done using the Python programming language, Keras application programming interface (API), and Tensorflow framework. Deep learning algorithms require large amounts of computing power, and typically running these algorithms on a system with only a Central Processing Unit (CPU) can take anywhere from hours to days. Graphics Processing Units (GPUs), which are more powerful than CPUs, are preferred when running deep learning algorithms since they can quickly compute the complex mathematical operations required by deep learning neural networks. GPUs can be expensive, therefore I utilized cloud GPU computing to run our deep learning models. GPU cloud computing reduced the overall training time for our models, allowing me to easily adjust model parameters and complete more runs to get thorough results. I obtained results by running all models using an Amazon Elastic Compute Cloud instance with the following configuration: 1 NVIDIA Tesla V100 GPU, 8 Intel Xeon E5-2686 v4 CPUs, 16 gigabyte (GB) GPU Memory, and a 100 GB solid state drive (SSD).

# CHAPTER SIX

# RESULTS

## Overall Performance Accuracy

The results of all implemented deep learning models and 2 traditional machine learning methods on the UCI HAR dataset are shown in Table 2, with the best performing model highlighted in bold [11]. The table lists the overall performance accuracy of each model in classifying the 6 activities, as well as the training time for the respective models. The same hyperparameters were applied to all the models, and they were all trained with the same data subset. Therefore, comparing the overall training time of each of these models gives us a better idea of how lightweight and efficient a model is. One drawback of deep learning algorithms can be the long training times, so obtaining a high accuracy with a low training time is compelling in the field of deep learning. From Table 2, we see that the 2-layer CNN combined with 1-layer LSTM outperformed all other models with a high accuracy of 94.7% and a training time of 7.7 minutes. All deep learning models outperformed the traditional methods presented by [5], which upholds that deep learning methods which automatically extract features and complete classification, outperform TML methods that require hand-crafted features [11].

Table 2. Summary of Performance Accuracy – UCI Dataset

| ***Summary of Performance Accuracy & Training Time*** | | |
|---|---|---|
| **Model** | **UCI HAR Accuracy (%)** | **Training Time (minutes)** |
| MC-SVM [5] | 89.3 | Unknown |
| MC-HF-SVM [5] | 89.0 | Unknown |
| 1-layer LSTM | 90.2 | 7.3 |
| 2-layer LSTM | 91.0 | 14.91 |
| 1-layer CNN | 91.1 | 3.2 |
| 2-layer CNN | 92.4 | 3.5 |
| 1-layer CNN-1-layer LSTM | 91.9 | 4.4 |
| 1-layer CNN-2-layer LSTM | 91.0 | 4.2 |
| **2-layer CNN-1-layer LSTM** | **94.7** | **7.7** |
| 2-layer CNN-2-layer LSTM | 94.3 | 6.7 |
| 2-layer CNN-3-layer LSTM | 92.5 | 9.3 |
| 3-layer CNN-1-layer LSTM | 91.6 | 4.9 |
| 3-layer CNN-2-layer LSTM | 91.7 | 7.4 |
| 3-layer CNN-3 layer LSTM | 92.9 | 9.9 |
| 4-layer CNN-1 layer LSTM | 93.8 | 7.1 |
| 4-layer CNN-2-layer LSTM | 92.5 | 7.2 |

S. Perez-Gamboa, Q. Sun and Y. Zhang, "Improved Sensor Based Human Activity Recognition via Hybrid Convolutional and Recurrent Neural Networks," *2021 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL)*, 2021, pp. 1-4, doi: 10.1109/INERTIAL51137.2021.9430460.

Based on the results of the UCI dataset, I noticed that models with an even number of CNN layers outperformed models with an odd number of CNN layers. These models were excluded from the next experiment with the UTD-MHA dataset. The results of all implemented deep learning models and 1 traditional machine learning method on the UTD-MHA dataset are shown in Table 3, with the best performing model being the 4-layer CNN-1-layer LSTM hybrid model with an accuracy of 92.94% for 15 activities, and 88.04% for 27 activities. Again, all deep learning methods outperformed the traditional method presented in [6], which confirms that deep learning is better for HAR, given large datasets with highly correlated activities.

Comparing our LSTM and CNN-only models from both datasets, we can see that the CNN models not only outperform the LSTM; they take less time to train. This is expected due to the sequential dependencies that embody LSTM and other RNN architectures. As explained in the methodology section, each time-step of an input sequence passed through an LSTM will be processed through gates to determine what information to keep in the memory cell. Each evaluation depends on the completion of previous steps, resulting in slower performance.

Table 3. Summary of Performance Accuracy – UTD-MHA Dataset

| ***Summary of Performance Accuracy & Training Time*** | | | | |
|---|---|---|---|---|
| **Model** | **Accuracy - 15 Activities (%)** | **Training Time (minutes)** | **Accuracy - 27 Activities (%)** | **Training Time (minutes)** |
| CRC [6] | Unknown | Unknown | 67.20 | Unknown |
| 1-layer LSTM | 88.42 | 5.36 | 84.55 | 9.82 |
| 2-layer LSTM | 90.25 | 16.48 | 87.27 | 17.74 |
| 1-layer CNN | 90.25 | 0.81 | 65.92 | 2.65 |
| 2-layer CNN | 91.53 | 0.93 | 87.78 | 3.95 |
| 2-layer CNN-1-layer LSTM | 92.79 | 1.48 | 82.50 | 4.77 |
| 2-layer CNN-2-layer LSTM | 92.66 | 1.59 | 87.92 | 4.60 |
| 2-layer CNN-3-layer LSTM | 92.12 | 4.11 | 87.99 | 5.88 |
| **4-layer CNN-1 layer LSTM** | **92.94** | **2.08** | **88.04** | **7.04** |

## Performance Metrics

When using human activity datasets for training and testing deep learning models, it is important for the dataset to be balanced in terms of the number and type of activities. An imbalanced dataset could mean that the overall performance accuracy is influenced by one activity that has a high recognition rate but ignores an activity with a poor recognition rate. To ensure that our datasets are balanced when applied to our models, it is important to examine the precision score, recall score, and F1 score for each.

When our models are classifying the sensor data, we encounter the following possibilities: true positives ($TP$), true negatives ($TN$), false positives ($FP$), and false negatives ($FN$). $TP$ and $TN$ classifications occur when our model correctly predicts whether the input sequence is or is not an activity, while $FP$ and $FN$ classifications occur when our model makes an incorrect prediction.

A precision score denotes the ratio of an activity's $TP$ classifications to the total $TP$ and $FP$ classifications for that activity. A precision score answers the question: How many human activities were accurately recognized and labeled by the model? Precision ($P$) is calculated with the following:

$$P = \frac{TP}{TP + FP}$$

A recall score denotes the ratio of an activity's $TP$ classifications to all classifications of that activity. Recall answers the question: of all the actual sequences of a specific activity, how many did the model correctly label? Recall ($R$) is calculated with the following:

$$R = \frac{TP}{TP + FN}$$

An F1 score is the weighted average of the precision score and the recall score, and takes both false positives and false negatives into account. The F1 score equally considers all classifications, giving us a more accurate look at the performance of a model. F1 scores ($F1$) are calculated with the following:

$$F1 = 2 \times \frac{R \times P}{R + P}$$

Table 4 shows the average of each of these metrics for all our deep learning models on the UCI Dataset. It shows us that our 2-layer CNN-1-layer LSTM model had the highest overall metrics with a mean precision score of 95%, mean recall score of 95%, and mean F1 score of 95%, which correctly reflects its high overall performance accuracy of 94.7%

Table 4. Summary of Performance Metrics - UCI Dataset

| *Summary of Performance Metrics* | | | |
|---|---|---|---|
| **Model** | **Mean Precision (%)** | **Mean Recall (%)** | **Mean Recall (%)** |
| 1-layer LSTM | 91 | 90 | 90 |
| 2-layer LSTM | 91 | 91 | 91 |
| 1-layer CNN | 91 | 91 | 91 |
| 2-layer CNN | 93 | 93 | 92 |
| 1-layer CNN-1-layer LSTM | 91 | 91 | 91 |
| 1-layer CNN-2-layer LSTM | 91 | 91 | 91 |
| 2-layer CNN-1-layer LSTM | 95 | 95 | 95 |
| 2-layer CNN-2-layer LSTM | 94 | 94 | 94 |
| 2-layer CNN-3-layer LSTM | 93 | 93 | 92 |
| 3-layer CNN-1-layer LSTM | 92 | 92 | 92 |
| 3-layer CNN-2-layer LSTM | 92 | 92 | 92 |
| 3-layer CNN-3 layer LSTM | 93 | 93 | 93 |
| 4-layer CNN-1 layer LSTM | 94 | 94 | 94 |
| 4-layer CNN-2-layer LSTM | 93 | 93 | 93 |

S. Perez-Gamboa, Q. Sun and Y. Zhang, "Improved Sensor Based Human Activity Recognition via Hybrid Convolutional and Recurrent Neural Networks," *2021 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL)*, 2021, pp. 1-4, doi: 10.1109/INERTIAL51137.2021.9430460.

Table 5 shows the average of each of these metrics for all our deep learning models on the UTD-MHA Dataset. It shows us that our 4-layer CNN-1-layer LSTM model had one of the highest overall metrics, which correctly reflects its high overall accuracy for both 15 and 27 activities.

By comparing Tables 2, 3, 4, and 5, we can see that our models' overall accuracy is proportional to their performance metrics, which lets us know that our datasets were balanced.

Table 5. Summary of Performance Metrics – UTD-MHA Dataset

| ***Summary of Performance Metrics*** | | | | | | |
|---|---|---|---|---|---|---|
| **Model** | **Mean Precision (%) 15 Activities** | **Mean Recall (%) 15 Activities** | **Mean F1-Score (%) 15 Activities** | **Mean Precision (%) 27 Activities** | **Mean Recall (%) 27 Activities** | **Mean F1-Score (%) 27 Activities** |
| 1-layer LSTM | 90 | 88 | 88 | 86 | 85 | 85 |
| 2-layer LSTM | 91 | 90 | 90 | 89 | 88 | 87 |
| 1-layer CNN | 91 | 90 | 90 | 70 | 67 | 65 |
| 2-layer CNN | 92 | 92 | 92 | 88 | 88 | 88 |
| 2-layer CNN-1-layer LSTM | 93 | 93 | 93 | 83 | 82 | 82 |
| 2-layer CNN-2-layer LSTM | 93 | 93 | 93 | 88 | 88 | 88 |
| 2-layer CNN-3-layer LSTM | 92 | 92 | 92 | 87 | 88 | 87 |
| 4-layer CNN-1 layer LSTM | 93 | 93 | 93 | 89 | 88 | 88 |

## Top Performing Models

After considering the performance of all our models, we offer a closer look at the top-performing hybrid models for both datasets.

### 2-layer CNN-1-layer LSTM Hybrid Model for UC Irvine Dataset

When our deep learning models are being trained, the training data is presented in batches until the entire training subset is passed through. With each batch, our model makes predictions and based on the error of these predictions, the weights of the model are updated. A single iteration of all training data passing through the model is referred to as an “epoch”. The number of epochs for training deep learning models is usually large to allow the model to properly train by repeatedly seeing the training data and sufficiently updating its weights to minimize loss. As we increase the number of epochs, we expect the overall accuracy to increase as well. Figure 12 demonstrates the accuracy rate of our model over epochs as it was repeatedly training, and as it was testing with unseen testing data. We can see that our model’s performance as it is training is what we expect: as the number of epochs increases, so does the accuracy. We expect our training data to have a higher accuracy than the testing data since it is data that the model repeatedly sees after each epoch. The testing data will have a lower accuracy because it is unseen data that is passed through the model only once to make the final predictions.

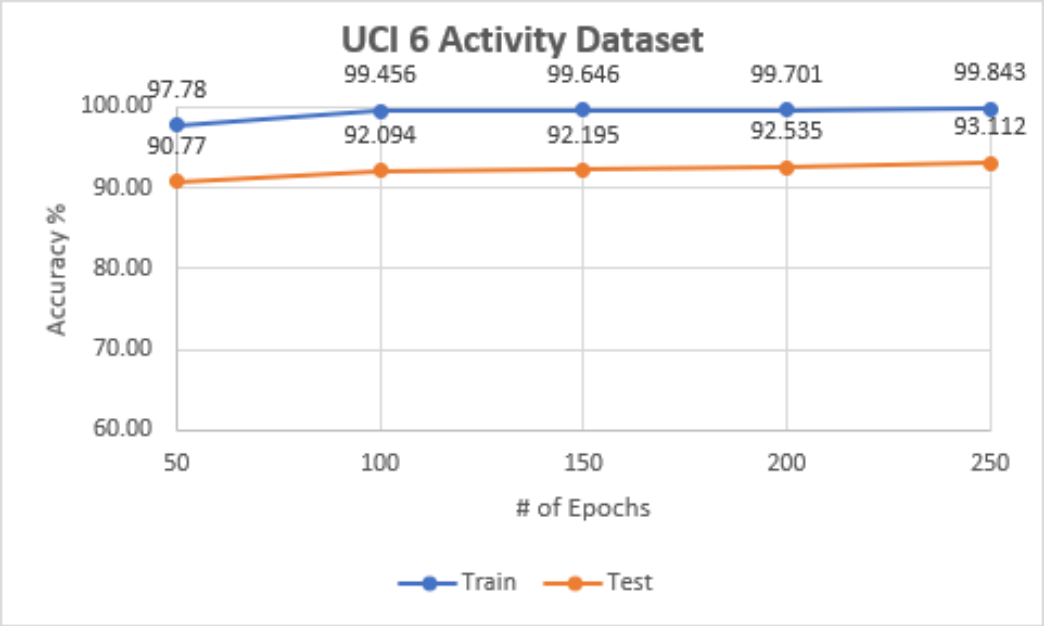

Figure 12. Accuracy vs Epochs for Training and Testing (UCI).

Figure 13 presents a confusion matrix for our model for each of the 6 activities in the dataset. The confusion matrix provides insight into the number of times our model classified a specific activity correctly or incorrectly classified it as another activity. Figure 13 shows that our model struggled the most with differentiating between the “sitting” and “standing” actions, this is likely due to the similarity in acceleration and subject orientation while performing these actions. Our model had a 100% correct classification rate for the “laying” activity, and only misclassified “walking downstairs” 3 times [11].

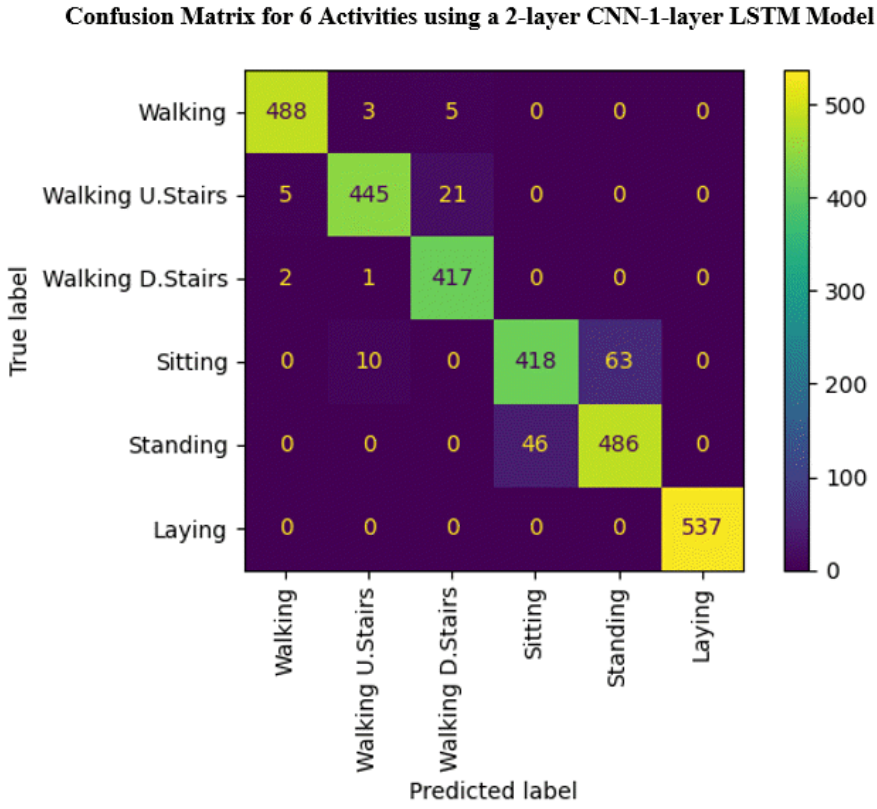


Figure 13. Confusion Matrix for UCI Dataset of 6 Activities.
S. Perez-Gamboa, Q. Sun and Y. Zhang, "Improved Sensor Based Human Activity Recognition via Hybrid Convolutional and Recurrent Neural Networks," *2021 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL)*, 2021, pp. 1-4, doi: 10.1109/INERTIAL51137.2021.9430460.

Figure 14 gives us a closer look at the precision score, recall score, and F1 score for each of the activities. By comparing Figure 13 and Figure 14, we can see that the results of the confusion matrix are directly proportional to the performance metrics.

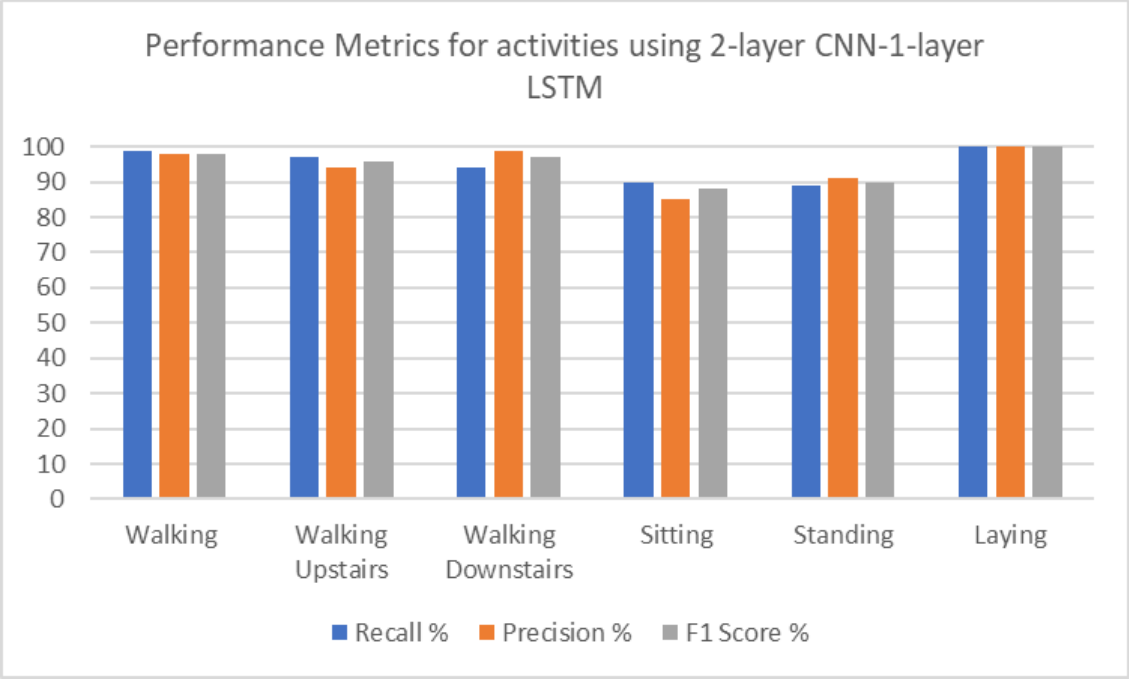


Figure 14. 2-layer CNN-1-layer LSTM Hybrid Model Performance Metrics. S. Perez-Gamboa, Q. Sun and Y. Zhang, "Improved Sensor Based Human Activity Recognition via Hybrid Convolutional and Recurrent Neural Networks," *2021 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL)*, 2021, pp. 1-4, doi: 10.1109/INERTIAL51137.2021.9430460.

4-layer CNN-2-layer LSTM Hybrid Model for UTD-MHA Dataset

Figure 15 demonstrates the accuracy rate of our 4-layer CNN-2-layer LSTM hybrid model over epochs as it was training and as it was testing with unseen data. We can see that although the training accuracy varies, the testing accuracy steadily increases as the # of epochs increases. This shows us that we could potentially increase the # of epochs to increase the overall performance, although this would mean that training time would also be increased.

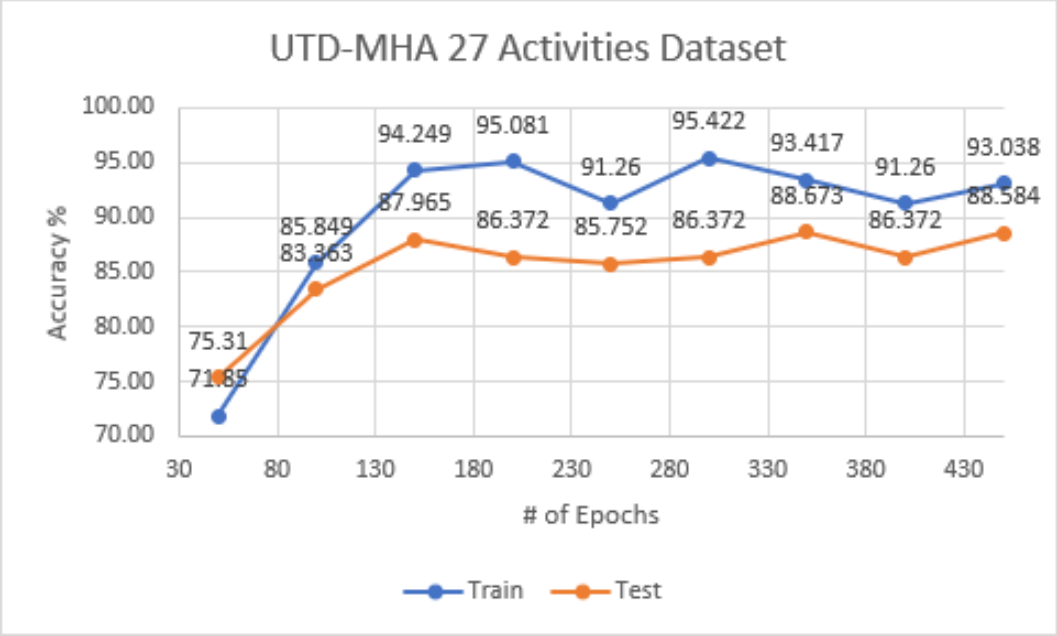


Figure 15. Accuracy vs Epochs for Training and Testing (UTD-MHA).

Figure 16 presents a confusion matrix for our model for each of the 27 activities in the dataset. This shows us that our model identified the “swipe right”, “clap”, “basketball shoot”, “squat”, and “lunge” actions quite well. It struggled the most classifying “throw”, “tennis swing”, and “pickup & throw”.

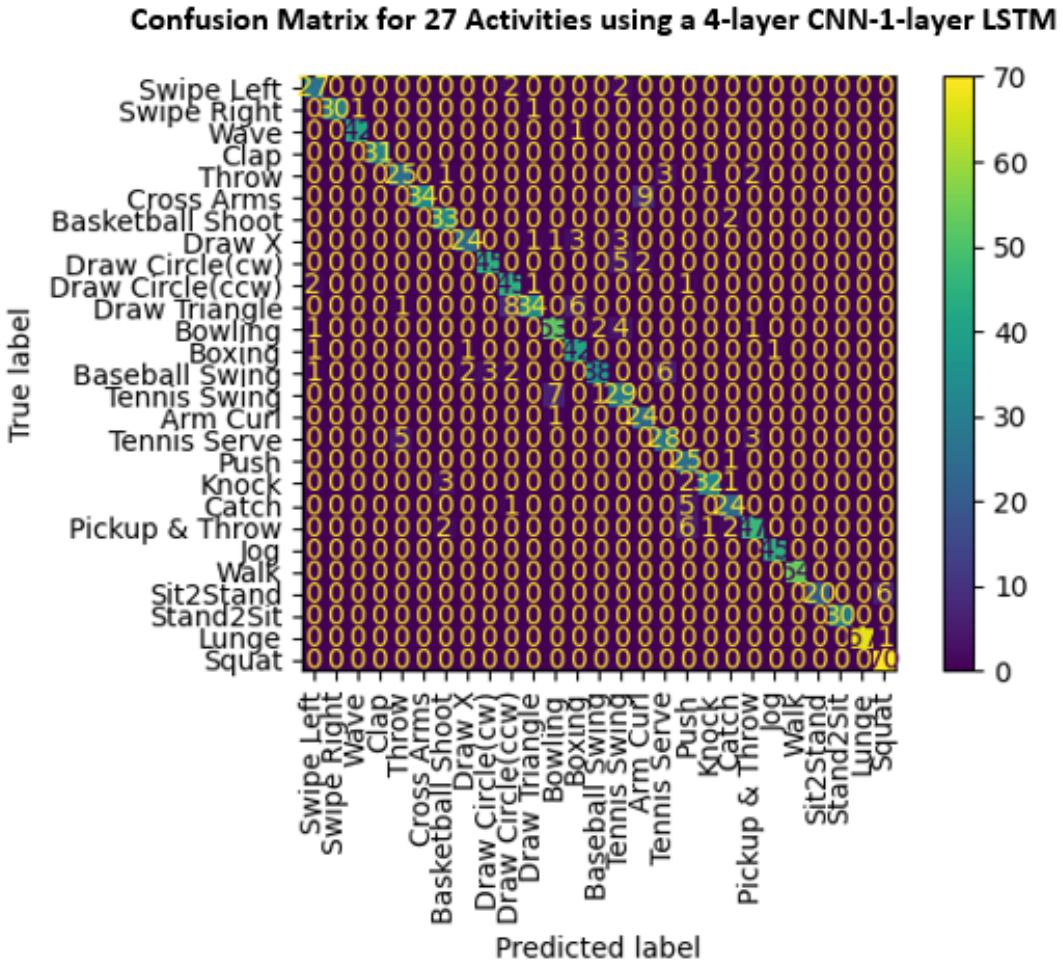


Figure 16. Confusion Matrix for UTD-MHA Dataset of 27 Activities.

Figure 17 gives us a closer look at the precision, recall, and F1 scores for each of the activities. We can see that “swipe right”, “clap”, “cross arms”, “jog”, “walk”, “sit2stand”, “stand2sit”, and “lunge” all had the highest recall score, meaning almost all sequences for those activities were correctly labeled. “Draw triangle”, “baseball swing”, and “draw x” had the lowest precision score, which means that the activities were incorrectly labeled by our model. As mentioned, the F1 score is one of the most useful ways to examine the classification accuracy of a model. The model performed best with “clap”, “walk”, and “stand2sit” with a 100% F1 score. The model struggled most with “tennis swing”, “push”, and “tennis serve”. By comparing Figure 16 and Figure 17, we can see that the results of the confusion matrix are directly proportional to the performance metrics.

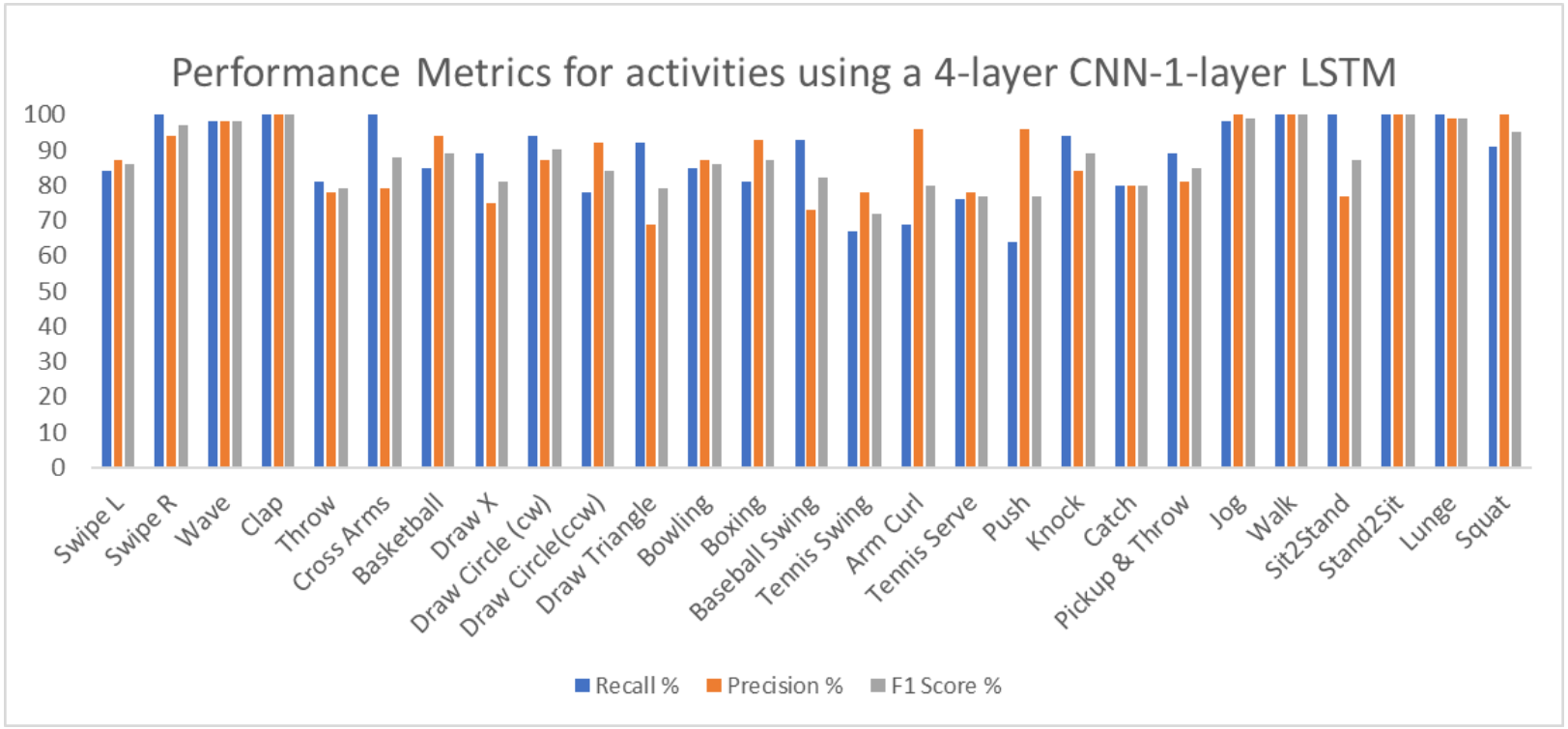


Figure 17. 4-layer CNN-1-layer LSTM Hybrid Model Performance Metrics

# CHAPTER SEVEN

# CONCLUSION

For this project, I explored the combination of deep learning models when analyzing two human activity datasets containing inertial sensor data. All presented deep learning models outperform previously presented TML methods. Our 2-layer CNN-1-layer LSTM hybrid model on the UCI dataset outperforms all other deep learning models with an accuracy of 94.7%. Additionally, our 4-layer CNN-1-layer LSTM hybrid model on the UTD-MHA dataset outperforms all other deep learning models with an accuracy of 92.94% for 15 activities, and 88.04% for 27 activities. Our presented lightweight hybrid models not only have a high-performance accuracy, but also have a fast model training time, which is compelling in the deep learning field. Overall, this project strengthens the premise that deep learning models are highly efficient in accomplishing sensor-based HAR and that there are simple ways to improve their performance through slight architecture adjustments. These models can be used for HAR across many domains such as health, fitness, social work, sociology, and gaming. There is further work that can be done in this area to strengthen performance, such as exploring different methods of sensor-based human data processing, optimizing deep learning model hyperparameters, and considering other deep learning models to combine. I hope to one day continue this research and apply it in other research fields.

REFERENCES

[1] Y. Hong, I. Kim, S. C. Ahn and H. Kim, "Activity Recognition Using Wearable Sensors for Elder Care," *2008 Second International Conference on Future Generation Communication and Networking*, pp. 302-305, 2008.

[2] J. Johnson et al., "A Wearable Mobile Exergaming System for Activity Recognition and Relaxation Awareness," *2019 IEEE International Systems Conference (SysCon)*, Orlando, FL, USA, pp.1-5, 2019.

[3] S. Oniga, A. Tisan and R. Bólyi, "Activity and health status monitoring system," *2017 IEEE 26th International Symposium on Industrial Electronics (ISIE)*, Edinburgh, pp. 2027-2031, 2017.

[4] R. Mohamed, T. Perumal, M. N. Sulaiman, N. Mustapha and M. N. Razali, "Conflict resolution using enhanced label combination method for complex activity recognition in smart home environment," *2017 IEEE 6th Global Conference on Consumer Electronics (GCCE)*, Nagoya, pp.1-3, 2017.

[5] D. Anguita, A. Ghio, L. Oneto, X.Parra and J. Reyes-Ortiz. "A Public Domain Dataset for Human Activity Recognition Using Smartphones," *21th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2013*. Bruges, Belgium 24-26 April 2013

[6] C. Chen, R. Jafari, and N. Kehtarnavaz, "UTD-MHAD: A Multimodal Dataset for Human Action Recognition Utilizing a Depth Camera and a Wearable Inertial Sensor"*, Proceedings of IEEE International Conference on Image Processing,* Canada, September 2015.

[7] J.-L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, D. Anguita, "Transition-aware human activity recognition using smartphones", *Neurocomputing,* vol. 171, pp. 754-767, Jan. 2016.

[8] H. Xu, Z. Huang, J. Wang and Z. Kang, "Study on Fast Human Activity Recognition Based on Optimized Feature Selection," *2017 16th International Symposium on Distributed Computing and Applications to Business, Engineering and Science (DCABES),* Anyang, pp. 109-112, 2017.

[9] N. F. Ghazali, N. Shahar, N. A. Rahmad, N. A. J. Sufri, M. A. As'ari and H. F. M. Latif, "Common sport activity recognition using inertial sensor," *2018 IEEE 14th International Colloquium on Signal Processing & Its Applications (CSPA)*, Batu Feringghi, pp. 67-71, 2018.

[10] Alom, Md. Zahangir & Taha, Tarek & Yakopcic, Chris & Westberg, Stefan & Sidike, Paheding & Nasrin, Mst & Hasan, Mahmudul & Essen, Brian & Awwal, Abdul & Asari, Vijayan, “A State-of-the-Art Survey on Deep Learning Theory and Architectures*,” Electronics 8*, no. 3:292, 2019.

[11] S. Perez-Gamboa, Q. Sun and Y. Zhang, "Improved Sensor Based Human Activity Recognition via Hybrid Convolutional and Recurrent Neural Networks," *2021 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL)*, 2021, pp. 1-4, doi: 10.1109/INERTIAL51137.2021.9430460.

[12] M.Zeng, L.T.Nguyen, B.Yu, O.J.Mengshoel, J.Zhu, P.Wu, J.Zhang , “Convolutional Neural Networks for Human Activity Recognition using Mobile Sensors”, *6th International Conference on Mobile Computing, Applications and Services*, pp.197-205, 2014.

[13] K. Nakano and B. Chakraborty, "Effect of dynamic feature for human activity recognition using smartphone sensors", *International Conference on Awareness Science and Technology*, pp. 539-543, 2017.

[14] I. Andrey, "Real-time human activity recognition from accelerometer data using convolutional neural networks", *Applied Soft Computing*, pp. 1-8, 2017.

[15] T. Yu, J. Chen, N. Yan and X. Liu, "A Multi-Layer Parallel LSTM Network for Human Activity Recognition with Smartphone Sensors," *2018 10th International Conference on Wireless Communications and Signal Processing (WCSP)*, Hangzhou, pp.1-6, 2018.

[16] N Y. Hammerla, S. Halloran and T. Ploetz, “Deep convolutional and recurrent models for human activity recognition using wearables,” *Cornell University arXiv*, 2016.

[17] F.J. Ordóñez,. D. Roggen, “Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition,” *Sensors*, *16*, 115, 2016.

[18] N. Tufek, M. Yalcin, M. Altintas, F. Kalaoglu, Y. Li and S. K. Bahadir, "Human Action Recognition Using Deep Learning Methods on Limited Sensory Data," in *IEEE Sensors Journal*, vol. 20, no. 6, pp. 3101-3112, 15 March, 2020.

[19] P. Rojanavasu, A. Jitpattanakul and S. Mekruksavanich, "Comparative Analysis of LSTM-based Deep Learning Models for HAR using Smartphone Sensor," *2021 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunication Engineering*, pp.269-272, 2021.

[20] B. Roher, "One Dimensional Convolutional Neural Networks," *End to End Machine Learning*, Feb-2020. [Online]. Available: https://e2eml.school/convolution_one_d.html

[21] J.B.Yang, M.N.Nguyen, P.P.San, X.L.Li,S. Krishnaswamy, "Deep Convolutional Neural Networks On Multichannel Time Series For Human Activity Recognition",*24th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 3995-4001, 2015.

[22] S. Hochreiter & J. Schmidhuber, "Long short-term memory," *Neural computation*, *9*(8), pp. 1735–1780, 1997.

[23] Y. Guan and T. Plötz, "Ensembles of deep LSTM learners for activity recognition using wearables," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technology*, vol. 1, no. 2, pp. 1–28, Jun. 2017.

[24] A. Dehgani, O. Sarbishei, T. Glatard and E. Shihab, "*A Quantitative Comparison of Overlapping and Non-Overlapping Sliding Windows for Human Activity Recognition Using Inertial Sensors"*, Sensors (Basel, Switzerland) vol. 19,22 5026. 18 Nov. 2019.