Project Report of Master of Engineering

Fall Detection Method based on Deep Learning using Accelerometer and Gyroscope Data

딥러닝 기반 가속도 및 자이로 센서 데이터 활용 낙상감지 방법

February 2022

Graduate School of Engineering Practice
Seoul National University
Department of Engineering Practice

Haesung Lee

Fall Detection Method based on Deep Learning using Accelerometer and Gyroscope Data

딥러닝 기반 가속도 및 자이로 센서 데이터 활용 낙상감지 방법

Professor Sungzoon Cho

Submitting a Master's Project Report

February 2022

Graduate School of Engineering Practice
Seoul National University
Department of Engineering Practice

Haesung Lee

Confirming the Master's Project Report written by
Haesung Lee
February 2022

Chair	Wooyoung Kwak
Examiner	Sungzoon Cho
Examiner	Yoonmo Koo

Abstract

As the world enters a super-aged society, fall accidents of elderly people are significantly increasing. These fall accidents, if not detected in time, may lead to serious consequences such as death in the worst cases. Therefore, when a fall accident occurs, it is necessary to establish a system for immediately detection. Among various methods for detecting falls, a device that is easy to wear and can be applied indoors and outdoors is devised. This study aims to develop a model that measures people movement using wearable-based accelerometer sensors and gyro sensors, analyzes acceleration and angular velocity, and classifies whether a fall occurs. In order to obtain data, an experiment was conducted in which 12 ADL movements and 4 Fall movements were repeatedly performed while the subjects were wearing a wearable device. ADL movements include sitting, standing, and walking, and the Fall movement consisting of falling forward and falling backward. In order to detect falls and non-falling, LSTM model of the Recurrent Neural Network (RNN) is used. The model was advanced through a data preprocessing and fine-tuning method applied to the input value of the LSTM model that determines whether to fall or not. In the experimental environment, the fall detection accuracy of the model is 99.91%, which is intended to determine the validity of fall detection from the perspective of

deep learning.

Keywords : Fall; 6-Axis sensor; Deep-Learning; Time series data; LSTM.

Student Number : 2020-20356

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Research Background and Objective

The average life span around the world is rapidly increasing due to the advancement in medical technology and the benefits of pharmaceutical industry. Korea's pace of population aging is faster than ever with the recent decline in fertility rates. This aging phenomenon is a social change that is evident mainly in developed countries rather than in underdeveloped countries.

The World Health Organization classifies a society as an aging society if the elderly population aged 65 or older accounts for more than 7% of the total population, an aged society if it accounts for more than 14%, and a super-aged society if it exceeds 20%. According to the National Statistical Office, Korea has already become an aging society as of 2018 with the elderly aged 65 or older accounting for 14.3% of the total population, and predicts to enter a super-aged society by 2025 with the elderly accounting for 20.3% of the total population.

An aging society does not simply mean an increase in a specific aging population, but implies that various social problems related to aging are occurring. The most representative social problem is social welfare issues related to treatment and care of increasing geriatric diseases and deterioration of health. According to the Health Insurance Major Statistics 2020 data published by the National Health Insurance Service of Korea, the total cost of health insurance in 2020 was 86.9 trillion won, of which 43% were the medical expenses for elderly aged 65 or older. Thus, maintaining the health of the elderly and extending their healthy life years are important tasks for the nation.

Fall is one of the main health problems in modern society with an aging demographic structure. Fall accidents are particularly problematic for the elderly as it takes a long time to recover from physical damage and dysfunction due to falls compared to young people, and decreased physical activity leads to rapid deterioration of health. Fall accidents can occur anytime, anywhere in everyday life, and, if not detected in time, secondary diseases may occur causing serious consequences such as death in the worst case. According to the World Health Organization, about 646,000 incidents of fatal falls occur every year worldwide, mainly in the elderly aged 65 or older [1].

Fall is an accident caused by a combination of various factors, so it is very difficult to prevent falls in advance. Therefore, in order to prevent injuries from exacerbating, the key to coping with fall accidents is to immediately detect them and promptly alert caregivers or institutions that can take emergency measures.

Previously studied fall detection methods can be generally divided into video-based posture classification methods and methods based on wearable devices with accelerometer sensors and angular velocity sensors. In the case of video-based fall detection methods, there is a spatial limitation because falls can only be detected in an environment where cameras installed indoors, and there is a risk of personal information exposure from videos. To overcome these limitations, this study aims to study a fall detection method using a deep learning LSTM model based on a small wearable device with high portability and activity.

1.2 Research Scope and Structure of Paper

In this study, various fall and activities of daily living(ADL)[1] are defined as 16 movements to detect fall accidents. To develop a model that determines the occurrence of fall, acceleration and angular velocity values of each movement are collected via a wearable device that includes an accelerometer sensor and a gyro sensor. The collected data is data with time series characteristics. We then use a binary classification model that classifies fall and non-fall, specifically the LSTM deep learning model from the research field of artificial intelligence. We tested the performance and results of LSTM using a confusion matrix.

This project report consists of five chapters. Chapter one introduces the background, purpose, and scope of the study. Chapter two deals with related works and deep learning models on the causes of falls and fall detection methods. In chapter three on the development of a fall detection model, we introduce experimental methods on data collection, modeling methods, and theoretical contents on model evaluation. In chapter four, we describe

[1] Activities of Daily Living (ADL) are physical movements that are repeated everyday to live independently, refers to movements such as feeding, dressing, communicating with others, and transportation.

model optimization methods including data preprocessing and fine tuning of the model. Finally, chapter five summarizes the results of the study and provides future research directions considering necessary matters.

Chapter 2

Background Knowledge and Related Research

2.1 Falls

A fall is defined as the unintentional movement of the body from a height of standing or lower to the floor or bottom surface, due to various causes such as an accident, fainting, convulsions, and paralysis [2]. Falls can generally occur in all ages but if it occurs in the elderly aged 65 or older, it may cause aftereffects that lead to death in severe cases.

Fall movements need to be divided into types based on the state of falling or posture. This is because falls can occur in various situations and positions such as while walking, standing, or sitting in a chair. El-Bendary (2013) classified falls into forward, lateral, and backward [3]. Other studies divided falls into a wider range, where the fall direction was also divided into front, side, and rear but the final movement when completely fallen was divided into lying down and sitting down. Physical damage and injuries areas due to falls are considered clinically different depending on the type and direction of falls. For example, a lateral fall has a high probability of femur cervical fracture, which greatly affects the quality of life as it is a body part that requires long-term rehabilitation [4].

Falls can occur from a combination of internal and external factors [4].

The intrinsic factors influencing fall accidents are age, disease, past fall experience, medication, cognitive impairment, emotional impairment, vision impairment, muscle weakness, balance impairment, and gait impairment. In addition, the external factors include slippery carpets, wet floors, and wet bathroom floors [5].

A study that surveyed 325 elderly people aged 60 or older who experienced falls reported that falls occurred due to a combination of multiple factors rather than a single factor [6]. As such, it is very difficult to predict falls in advance because falls are a kind of accident caused by complex causes. Thus, it is critical to promptly inform someone who can provide help such as immediately calling a caregiver or emergency service after the accident occurs.

2.2 Fall Detection Techniques

There are two main ways to detect falls. One is a method of detection by analyzing video information captured by the subject and the other is a method of detection by discriminating the movement using a sensor worn by the subject.

The former method of extracting a target object from recorded video and analyzing movement has a very high accuracy of 93% [7]. This method processes image data of a video using a signal processing technique. Therefore, the video-based fall detection is limited in that it can only detect falls that occur within the shooting range of the subject. In addition, the video must constantly monitor the subject leading to a possibility of infringement of personal information, and has the disadvantage of being vulnerable when it is difficult to secure a view, such as at night. Recently, a method of identifying human motion based on skeletal data is used in fall detection techniques [8]. These skeleton data are temporal and spatial data that change over time, and the authors studied a fall detection method that estimates the acceleration of changes in the positions of the head and shoulders based on human skeleton keypoints information extracted from PoseNet.

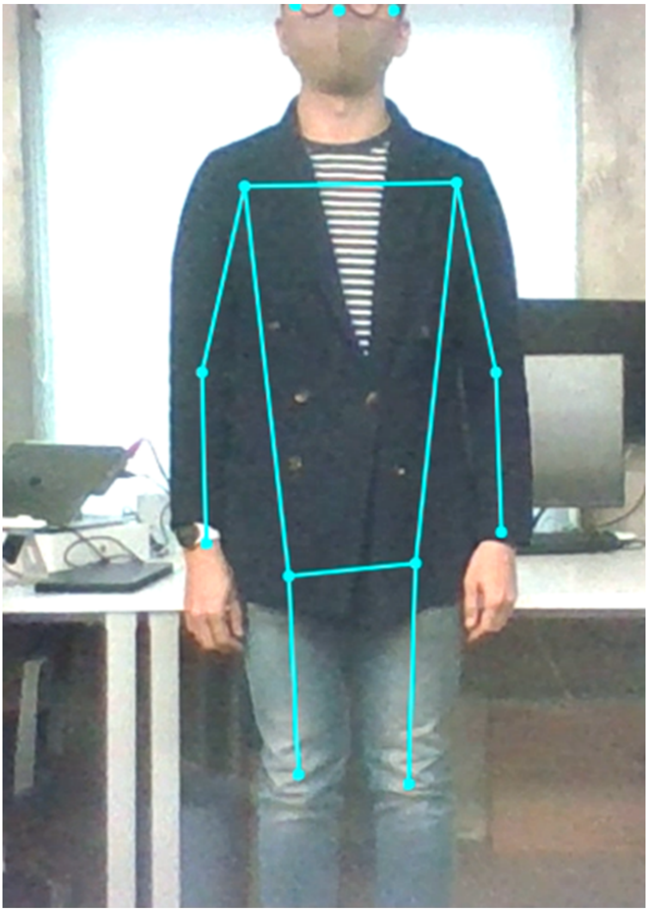


Figure 2.1: Posture detection using PoseNet.

Accelerometer sensors and gyroscope sensors are commonly used when utilizing sensors. The acceleration and angular velocity information of the subject wearing the sensors are analyzed to determine the occurrence of fall. Compared to the video-based fall detection method, the fall detection method with sensors has the less risk of infringement of personal information and has the advantage of being able to detect falls regardless of where they occur [2].

There is a method of using smartphones with built-in accelerometer sensors. This method analyzes thresholds of signal vector magnitude or signal magnitude area through acceleration information to determine fall occurrences [9].

2.3 Machine Learning

Machine learning is a research field in which statistics, artificial intelligence, and computer science are intertwined, and is also referred to as predictive machine learning. Depending on how it learns, it is divided into supervised learning, unsupervised learning, and reinforcement learning [10].

Ensemble is a technique that combines several learning algorithms to create a stronger model. The types of ensemble techniques are voting, bagging, boosting, and stacking. Bagging replicates several samples of training data, configures them slightly differently, creates a weak model with a weak learner for each replica of training data, and then combines models into one.

Random Forest is the most effective and widely used model among bagging-based learning algorithms. Random forest used a modified tree learning algorithm to randomly select partial data for the feature at each segmentation step and inspect it, to eliminate the correlation of the tree. If several features have a decisive effect on the target, the samples are divided into several trees based on these features. Then, trees with high correlation form a forest. Random forests are effective because multiple samples of the original dataset can be used to reduce the variance of the final model.

2.4 Recurrent Neural Networks and LSTM

The recurrent neural network (RNN) is a deep learning model mainly used to process data reflecting sequential attributes. Data of sequential attributes means that objects in the dataset have an order. Examples of sequential data include voice, sentence, seismic waves, DNA sequence, and time-series data. RNN can analyze these sequential data because of its recursive structure that determines the output value of the current input value through the previous input value.

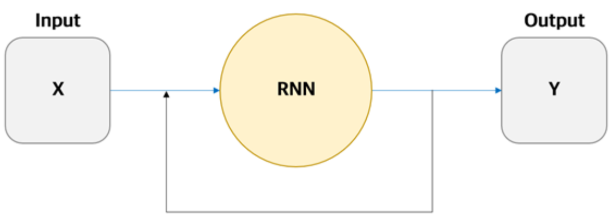


Figure 2.2: Structure of RNN.

The recurrent neural network (RNN) is a deep learning model mainly used to process data reflecting sequential attributes. Data of sequential at-

tributes means that objects in the dataset have an order. Examples of sequential data include voice, sentence, seismic waves, DNA sequence, and time-series data. RNN can analyze these sequential data because of its recursive structure that determines the output value of the current input value through the previous input value.

Due to the structure of RNN models, if the interval between input and output data is prolonged, the long-term dependency problem limits the model from optimizing. That is, when the RNN model learns based on the gradient descent method, a vanishing gradient problem or exploding gradient problem occurs resulting in poor learning performance [11].

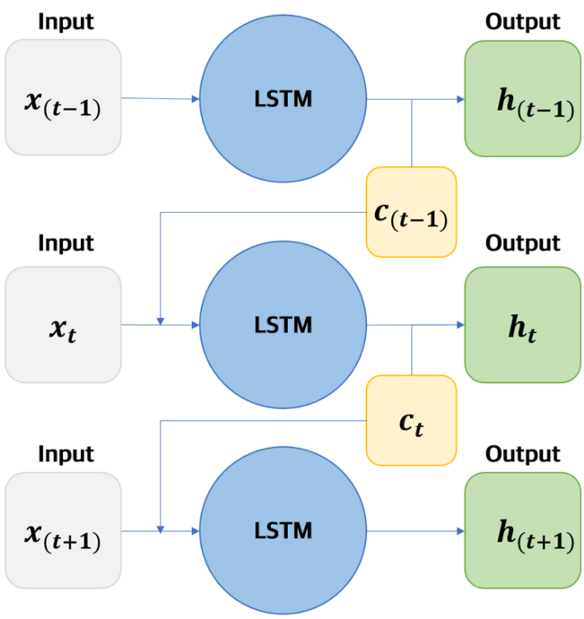


Figure 2.3: Structure of LSTM.

Long Short-Term Memory (LSTM) networks are designed to deal with the long-term dependence problem of RNN. LSTMs are characteristic in that it exists based on a cell state including the input gate, forget gate, and output gate [12].

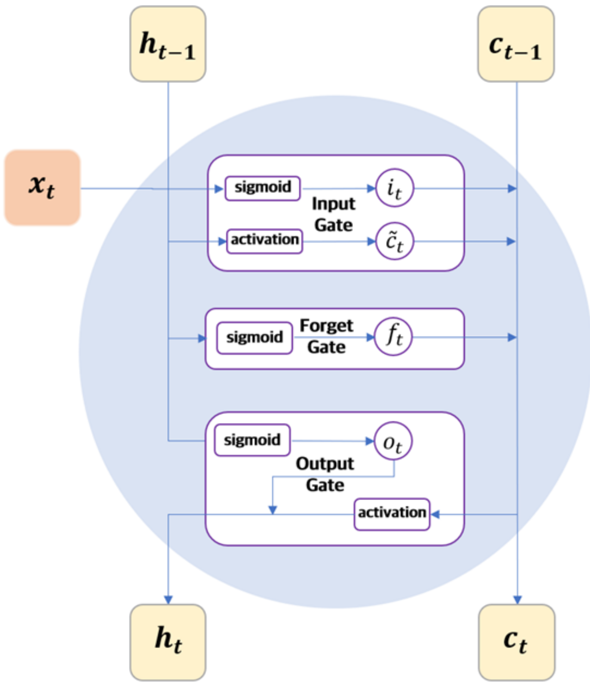


Figure 2.4: Structure of LSTM cell.

The cell state C_t is a factor that serves to transfer past information and is determined by the current cell state information that is determined by information transmitted from the previous cell state C_{t-1} and the gate. The current cell gate, cell state, and output operations consist of a linear combination of the weights, bias, and connection of the previous cell output and the current input $[h_{t-1}, x_t]$, which is then passed through the gate and the activation function. This can be expressed as equations 2.1 2.6 below. As for the activation function, hyperbolic tangent or soft sign functions may be used as expressed by equations 2.7 and 2.8. A hyperbolic tangent was used in this study.

$$i_t = \sigma\left(W_i\left[h_{t-1}, x_t\right] + b_i\right) \tag{2.1}$$

$$f_t = \sigma\left(W_f\left[h_{t-1}, x_t\right] + b_f\right) \tag{2.2}$$

$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right) \tag{2.3}$$

$$\widetilde{C_t} = \text{activation}\left(W_c\left[h_{t-1}, x_t\right] + b_c\right) \tag{2.4}$$

$$C_t = f_t \times C_{t-1} + i_t \times \widetilde{C_t} \tag{2.5}$$

$$h_t = o_t \times \text{activation}\left(C_t\right) \tag{2.6}$$

$$\tanh(x) = \tfrac{\sinh(x)}{\cosh(x)} = \tfrac{e^x - e^{-x}}{e^x + e^{-x}} \tag{2.7}$$

$$\text{softsign}(x) = \tfrac{x}{1+|x|} \tag{2.8}$$

Chapter 3

Methods

3.1 Measurement Methods and Devices

In this study, subjects wear a wearable device that measures acceleration and angular velocity data for determining actions and movements. The wearable device consists of STM32L MCU (ST Microelectronics) which allows operation on low power, and six-degree BMI 160 IMU sensor (Bosch) which can measure three-axis acceleration and three-axis angular velocity. The device has a Li-polymer type battery so that it could be used without a separate power supply, and the size and weight of the device is manufactured at a comfortable level for subjects. The sampling rate is set to 33Hz to measure acceleration (m/s^2) and angular velocity (deg/s) every .0303 seconds. The collected data is used to analyze movements and determine fall.

Item	Type	Specifications
Processor	MCU	STM32L, Arm 32-bit, Cortex-M4
Memory	Flash Memory RAM	Serial Flash, 8Mb
Location Determination		GPS, GLONASS
Sensor	Accelerometer & Gyroscope(6-Axis)	BMI160, BOSCH
Power	Operating Voltage	3.3 $\sim$ 4.3V
	Rating Voltage	3.8V
Battery	Li-Polymer	500mAh
Operating Temperature		-20 $\sim$ 50 °C
Exterior	Size	52mm * 75.5mm * 20.4mm
	Weight	56g

Table 3.1: Wearable Sensor Specification.

(a) Front

(b) Back

Figure 3.1: Wearable device.

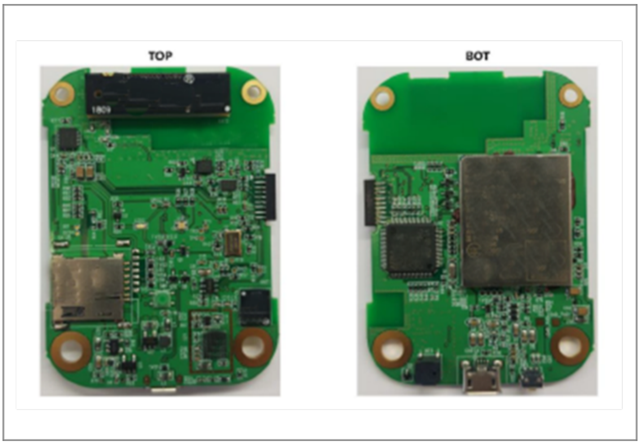


Figure 3.2: Wearable device PCB.

3.2 Definition of Falls and Daily Living Activities

We predefine activities of daily living and fall movements with references to previous work on ‘Mobiact & Mobifall’ and ‘Smartfall’ [13, 14]. Activities of daily living are divided into 12 movements including standing, walking, jogging, jumping, stairs up, stairs down, stand to sit, sitting on chair, sit to stand, car step in, car step out, and lying. Fall movements consist of four movements including forward lying, front knees lying, back sitting chair, and sideward lying.

No.	Class	Case
1	ADL	Standing
2		Walking
3		Jogging
4		Jumping
5		Stairs up
6		Stairs down
7		Stand to sit
8		Sitting on chair
9		Sit to stand
10		Car-step in
11		Car-step out
12		Lying
13	Fall	Forward-Lying
14		Front-Knees-Lying
15		Back-Sitting-Chair
16		Sideward-Lying

Table 3.2: Definition of fall movements and ADL.

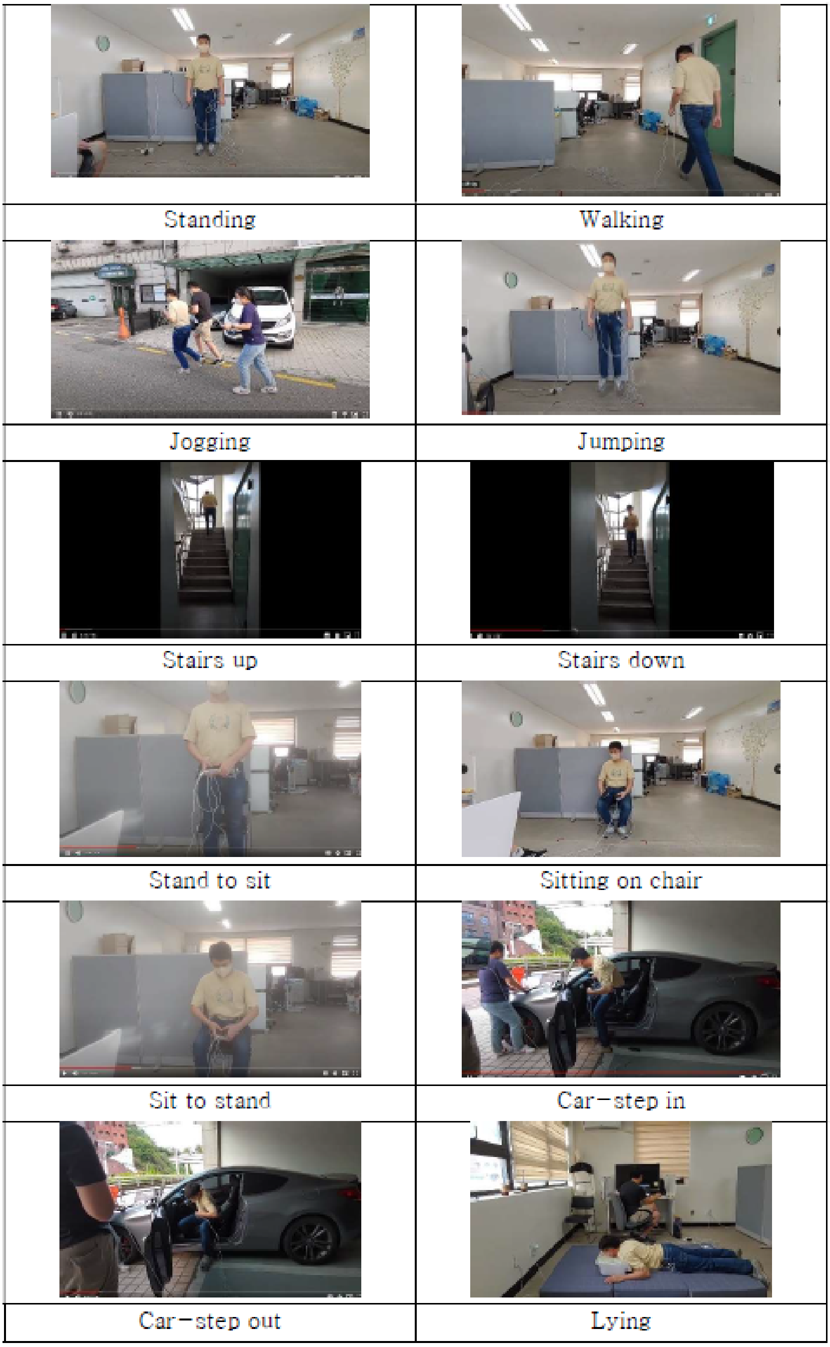


Figure 3.3: ADL movements.

Figure 3.4: Fall movements.

3.3 Development of Fall Detection Model

To detect falls, we collect time-series data on acceleration and angular velocity of the subject's movements and develop a fall detection model trained on the dataset with LSTM among deep learning models. Figure 3.5 illustrates the structure of the fall detection system. Subject data are collected with accelerometer and gyroscope sensors which includes time, acceleration tri-axial, and angular velocity trial-axial information. Data is preprocessed and divided into train and test sets. Finally, a model classifying fall and non-fall movements is completed.

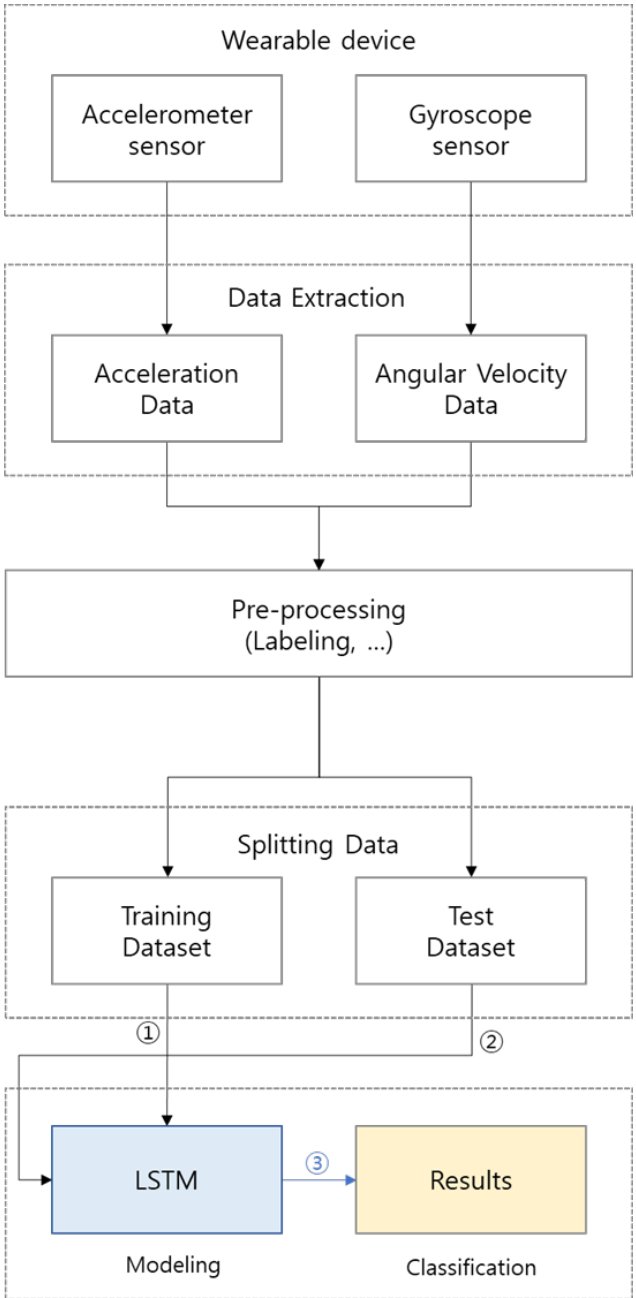


Figure 3.5: Structure of fall detection system using LSTM.

3.4 Performance Evaluation Metrics

We used a confusion matrix of model predictions and actual data values to evaluate the performance of the binary classification model classifying fall and non-fall.

Accuracy is a commonly used metric that assesses how correct the model is in classifying input data. However, it is limited in evaluating the overall performance of the model when true positives and true negatives are imbalanced. In this case, precision and recall are used to complement the accuracy metric. Precision is the ratio of actual cases that are correctly classified over positively classified cases. Recall is the ratio of actual cases that are correctly classified over all actual cases. F1 score is the harmonic average of precision and recall.

Fall accidents in elderly may result in critical physical damage, the model should minimize cases classifying fall as non-fall (false negatives). Thus, this study includes accuracy, precision, recall, and F1 score as performance metrics.

		Actual Class	
		Positive(Fall)	Negative(ADL)
Predicted Class	Positive(Fall)	True Positive(TP)	False Positive(FP)
	Negative(ADL)	False Negative(FN)	True Negative(TN)

Table 3.3: Confusion Matrix.

$$\text{Accuracy } = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.1}$$

$$\text{Precision } = \frac{TP}{TP + FP} \tag{3.2}$$

$$\text{Recall } = \frac{TP}{TP + FN} \tag{3.3}$$

$$\text{F1 Score } = 2 \times \frac{\text{Precision } \times \text{ Recall}}{\text{Precision } + \text{ Recall}} \tag{3.4}$$

Chapter 4

Results

4.1 Data Collection

Two subjects are required to perform fall movements and daily living activities while wearing the wearable device. Subjects are physically healthy male adults with low risk of injuries from the experiment, and fall movements are performed on a mattress to prevent injuries. Before the experiment, subjects are notified of possible risk of injuries from the experiment.

Figure 4.1: Environment of fall experiment.

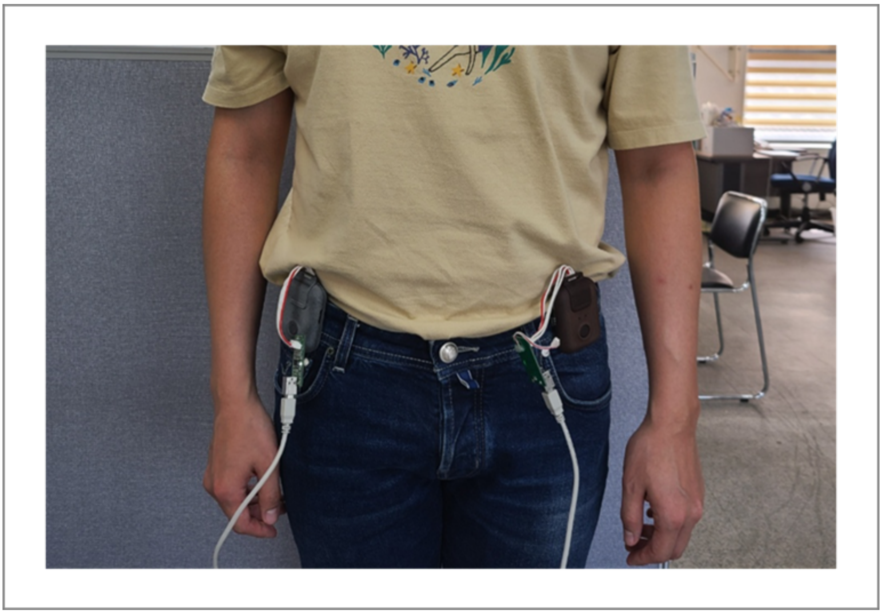


Figure 4.2: Example of wearing the wearable device.

As in Figure 4.2, a wearable device for data collection is attached to the subject's waist. The device is connected to a computer with USB cables to transmit and save data onto the computer. For reliability of data, data is simultaneously collected from two wearable devices with equal specifications.

Data on time (in seconds), three-axis acceleration, and three-axis angular velocity is collected 33 times per second and transmitted to a USB-connected computer through Tera Term[2] in a serial communication method. In addition, we record videos of subjects performing ADL and fall movements in order to annotate movements in the data preprocessing stage.

[2]Tera Term is an open-source free terminal emulator program developed by Takashi Teranishi in Japan and is used for serial communication purposes.

Item	Unit
Time stamp	sec
3-axis Acceleration (Acc_x, Acc_y, Acc_z)	m/s^2
3-axis Angular velocity (Gyro_x, Gyro_y, Gyro_z)	deg/s

Table 4.1: Data collection items and units.

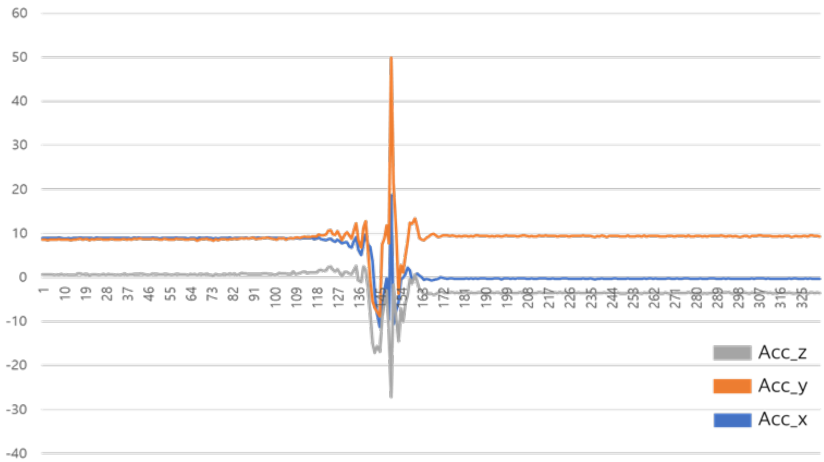
The lying movement of ADL is measured using data collected during timepoints after a fall movement.

Class	Case	Subject(People)	Device(EA)	Trials	Duration(sec)
ADL	Standing	2	2	2	400
	Walking	2	2	2	400
	Jogging	2	2	1	240
	Jumping	2	2	1	240
	Stairs up	2	2	15	10
	Stairs down	2	2	15	10
	Stand to sit	2	2	15	10
	Sitting on chair	2	2	1	240
	Sit to stand	2	2	15	10
	Car-step in	2	2	15	10
	Car-step out	2	2	15	10
	Lying	2	2	32	
Fall	Forward-Lying	2	2	8	10
	Front-Knees-Lying	2	2	8	10
	Back-Sitting-Chair	2	2	8	10
	Sideward-Lying	2	2	8	10

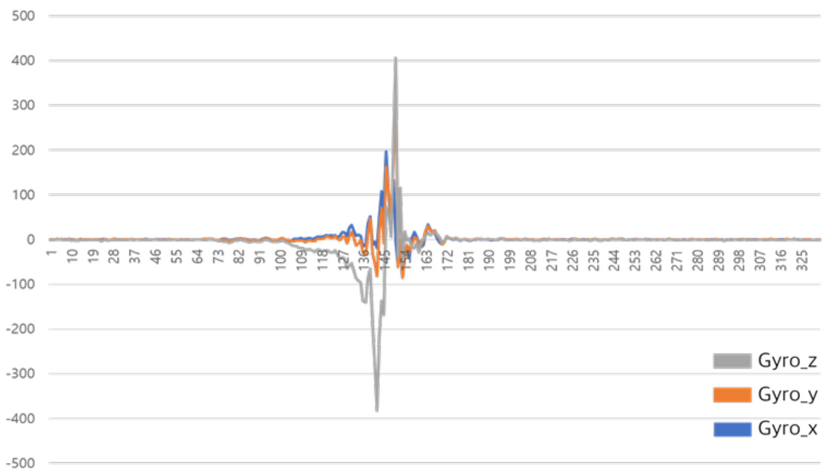
Table 4.2: Experimental history.

4.2 Data Preprocessing

Figures 4.3 to 4.6 are four types of fall graphs visualizing characteristics of fall data. We identify instantaneous changes in acceleration and angular velocity when falls occur.

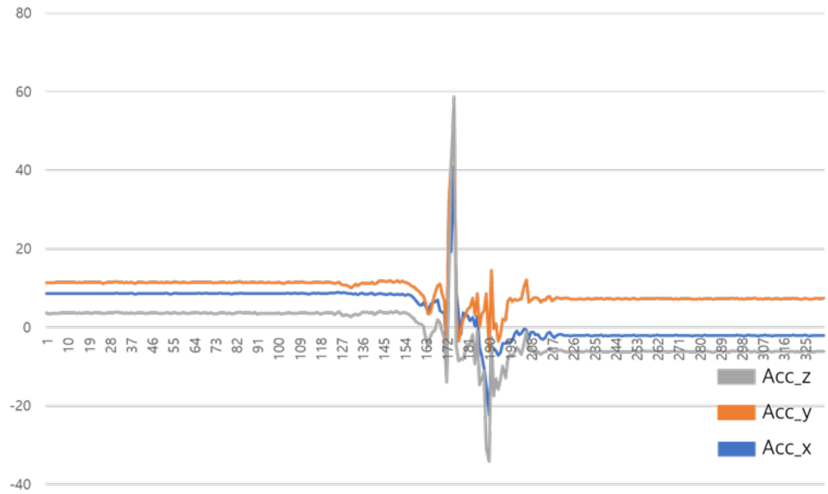


(a) Acceleration

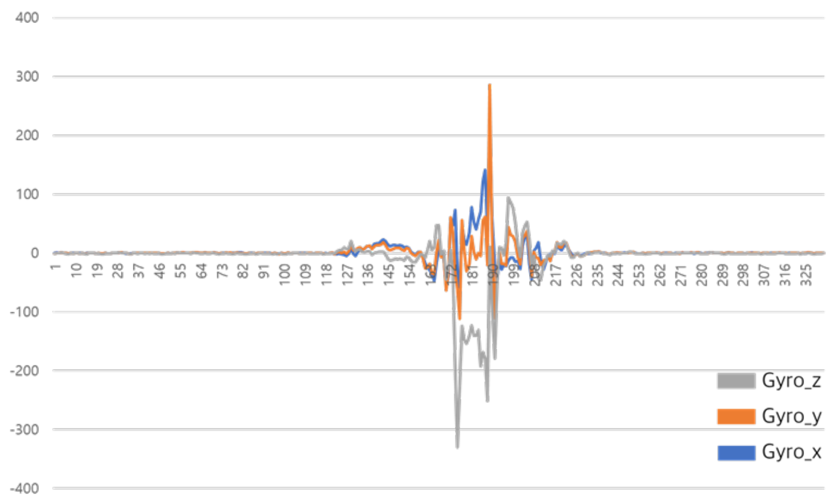


(b) Angular velocity

Figure 4.3: Graph of FOL (Forward-Lying) fall.

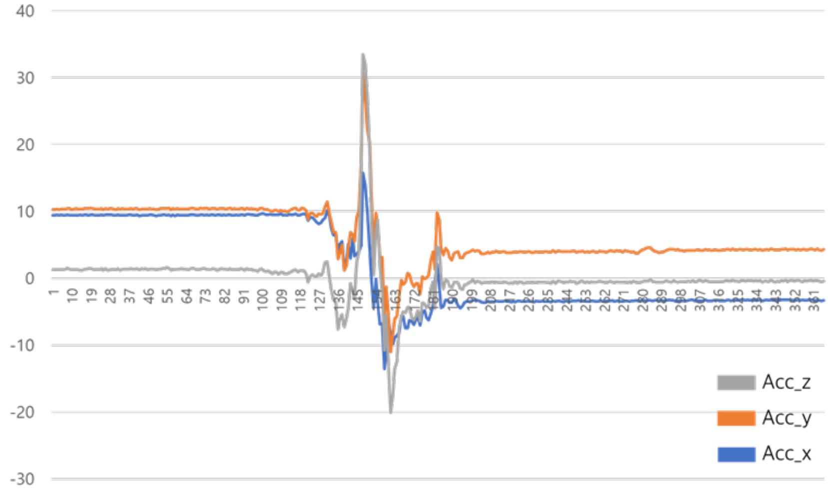


(a) Acceleration

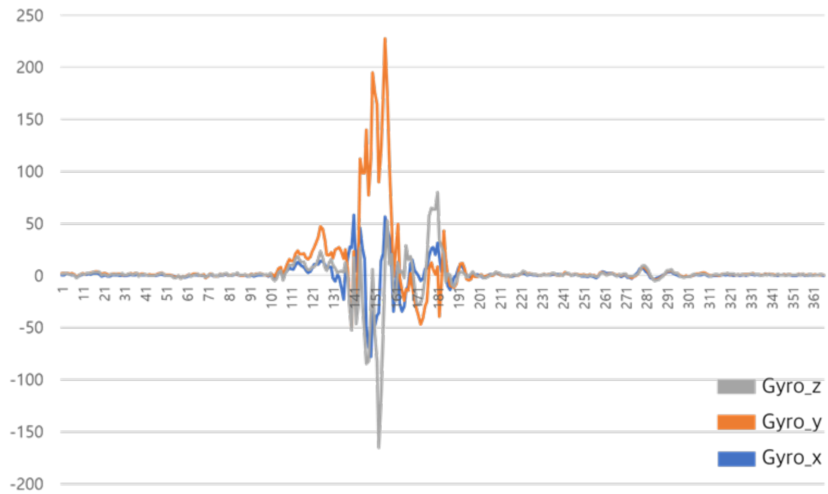


(b) Angular velocity

Figure 4.4: Graph of FKL (Front-Knees-Lying) fall.

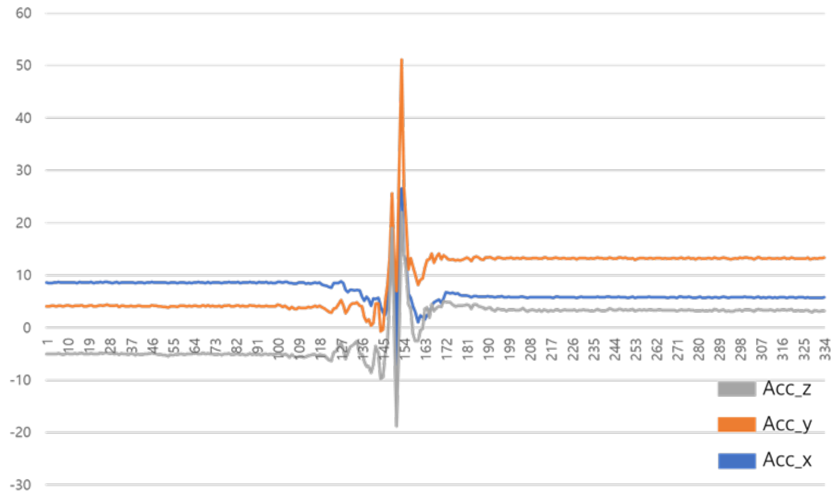


(a) Acceleration

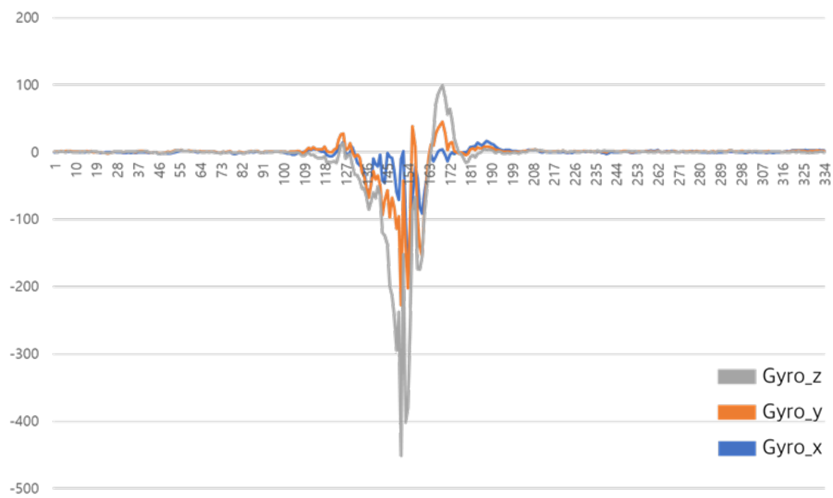


(b) Angular velocity

Figure 4.5: Graph of BSC (Back-Sitting-Chair) fall.



(a) Acceleration

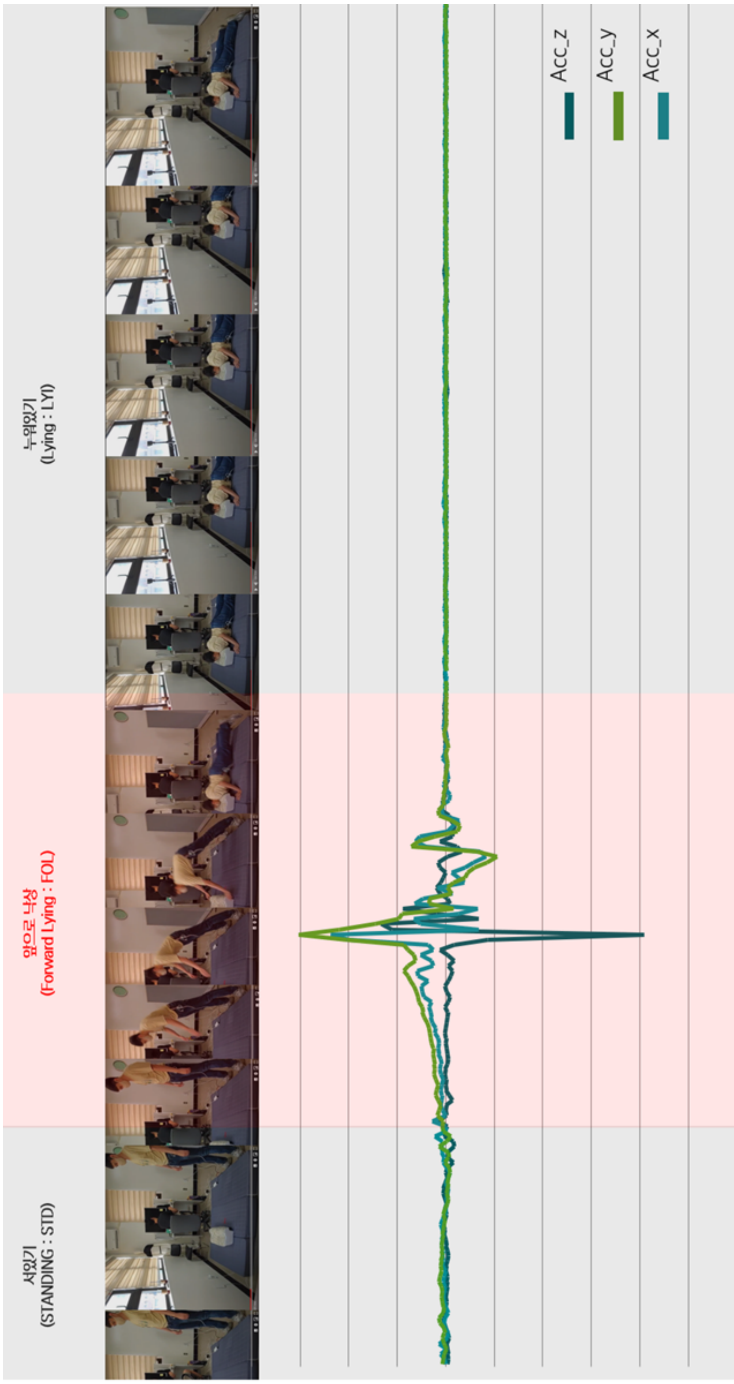


(b) Angular velocity

Figure 4.6: Graph of SDL (Sideward-Lying) fall.

Before and after the actual fall motion include standing and lying movements in which acceleration and angular velocity converge to 0 as there are no movements during these periods. Of the ten seconds of a FOL (Forward-

Lying) movement, only two seconds are actual timepoints of a FOL fall movement data. Thus, we label data to distinguish start and end timepoints of STD (Standing), FOL (Forward-Lying), and Lying (LYI) fall movements. The data annotation was performed by comparing the collected data to video frames of subjects conducting the movements during the experiment, and labeling start and end timepoints.



(a) Video frames with graphs

rel_time	acc_x	acc_y	acc_z	gyro_x	gyro_y	gyro_z	label
0	9.3	−2.39	−8.73	0.37	0.79	1.71	STD
0.031	9.21	−2.59	−8.62	0.31	0.49	2.38	STD
0.062	9.11	−2.61	−8.64	−0.79	1.53	2.69	STD
0.093	9.24	−2.54	−8.58	−1.22	0.98	1.53	STD
0.124	9.18	−2.73	−8.72	−1.59	1.22	2.08	STD
0.155	9.14	−2.65	−8.62	−1.89	0.61	1.1	STD
0.186	9.24	−2.72	−8.71	−1.71	1.4	0.73	STD
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
4.557	5.42	−2.48	−8.37	1.77	47.42	56.76	FOL
4.588	3.88	−3.29	−8.49	−2.32	55.73	59.75	FOL
4.619	2.09	−3.01	−8.31	5.49	46.2	62.56	FOL
4.65	1.66	−2.68	−8.01	16.05	23.01	57.8	FOL
4.681	1.49	−2.24	−8.57	22.52	19.17	52.86	FOL
4.712	0.19	−2.08	−9.79	13.37	24.17	65.12	FOL
4.743	0.58	0.25	−9.27	25.27	14.47	56.82	FOL
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
9.827	−1.55	−7.92	−3.69	2.08	0.18	−0.98	LYI
9.858	−1.61	−7.83	−3.76	1.46	−0.24	−0.31	LYI
9.889	−1.48	−7.83	−3.69	1.77	0.43	0	LYI
9.92	−1.56	−7.81	−3.71	1.59	0.67	−0.37	LYI
9.951	−1.58	−7.81	−3.73	1.83	0.18	−0.49	LYI
9.982	−1.57	−7.75	−3.64	1.28	0.12	−0.79	LYI
10.013	−1.64	−7.81	−3.73	1.65	0	0	LYI

(b) Raw data and annotation forms

Figure 4.7: Annotation of fall movements based on comparison with video frames.

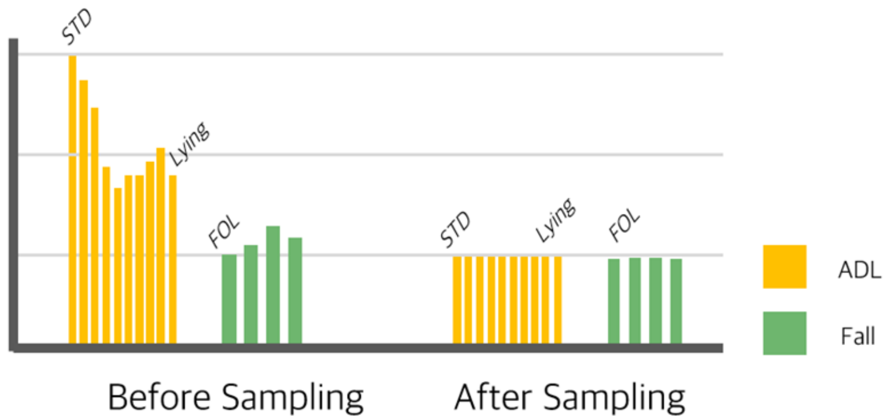


Figure 4.8: Under-sampling method.

As the amount of ADL data and fall data are imbalanced, the model could train on biased data resulting in low performance [15]. Thus, we use under-sampling matching the larger ADL data to fall data to balance the data.

Class	Case	Number of Data	
		Before Sampling	After Sampling
ADL	Standing	121,357	2,065
	Walking	107,732	2,065
	Jogging	33,191	2,065
	Jumping	32,196	2,065
	Stairs up	30,065	2,065
	Stairs down	30,846	2,065
	Stand to sit	19,450	2,065
	Sitting on chair	32,266	2,065
	Sit to stand	19,363	2,065
	Car-step in	20,300	2,065
	Car-step out	20,098	2,065
	Lying	19,814	2,065
Fall	Forward-Lying	2,065	2,065
	Front-Knees-Lying	3,121	2,065
	Back-Sitting-Chair	3,051	2,065
	Sideward-Lying	2,644	2,065

Table 4.3: Number of data pre and post sampling.

4.3 Model Fine-Tuning

As parameters for fine tuning the LSTM model, we use window size, learning rate, and number of epochs. First, window size should be an appropriate size in which the repeated periods and characteristic values of the pattern of the time series data are included. Thus, we compare performance of models with window size of $1/3$ secs, $2/3$ secs, 1 secs, 2 secs, and 3 secs.

Window Size	**Accuracy**
$1/3$ sec	0.9055
$2/3$ sec	0.9558
1 sec	0.9899
2 sec	0.9943
3 sec	0.9422

Table 4.4: Accuracy depending on window size.

If the learning rate is too big, the model may not train. If the learning is too small, training will take too long and may find the local minima instead of the global minima. In this study, we compare learning rates of .0001, .001, and .01.

Learning Rate	**Accuracy**
0.0001	0.9573
0.001	0.9991
0.01	0.9945

Table 4.5: Accuracy depending on learning rate.

Lastly, epochs refer to the exhaustive training iterations on the dataset which is set to 10 for this LSTM model. To sum up, optimized parameter values via fine-tuning are window size of 2 secs, learning rate of .001, and epoch of 10.

Item	Value
Window Size	2 sec
Learning Rate	0.001
Epoch	10

Table 4.6: Optimization of LSTM model.

4.4 Performance and Results Analysis

The fall detection LSTM model showed an evaluation index accuracy of .9991, recall of .9981, precision of .9983, and F1 score of .9982. Accuracy reached .999 at epochs of three, and thereafter, the performance of the model saturated to 99.9% without significant improvements depending on epochs.

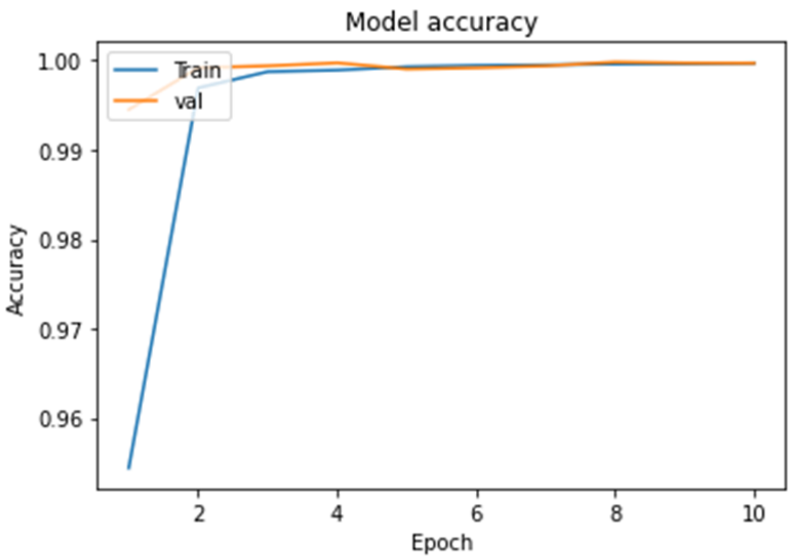


Figure 4.9: LSTM accuracy depending on epochs.

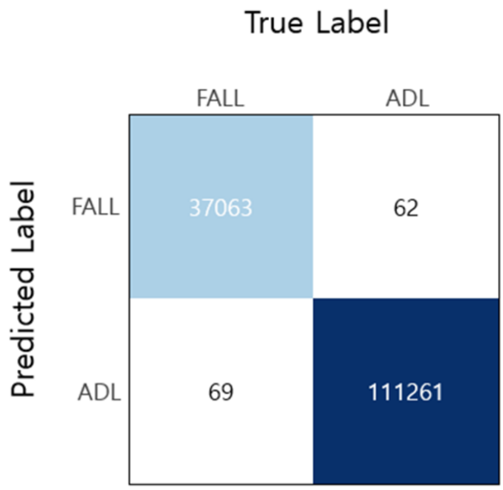


Figure 4.10: Confusion Matrix of LSTM Model.

The fall detection LSTM model using wearable device-based accelerometer and angular velocity sensor could determine the occurrence of falls with high accuracy. In particular, it showed the best performance when the window size was 2 seconds after fine-tuning the model. Therefore, the appropriate length of the cycle of patterns that include singularities of each ADL and fall movements was identified to be 2 seconds on average.

Model	F1 Score	Precision	Recall	Accuracy
LSTM	0.9982	0.9983	0.9981	0.9991
Decision tree	0.7388	0.7328	0.7448	0.8689
Random Forest	0.8368	0.8124	0.8627	0.9162
K-NN	0.7958	0.8007	0.7910	0.8990

Table 4.7: Experimental Results.

Lastly, the LSTM's performance reached F1 score of 99.82%, precision of 99.83%, recall of 99.81%, and accuracy of 99.91%. This demonstrated that the LSTM model shows higher performance than machine learning-based decision trees, random forest, and K-NN models, and that utilizing the LSTM model as a fall detection method is an excellent strategy.

Chapter 5

Conclusion

5.1 Discussion

This study examined a fall detection model of deep learning LSTM using data collected from accelerometers and gyro sensors. We developed a wearable device to collect training data and performed fall and activities of daily life(ADL) movement experiments on two subjects. The collected fall data include a standing state and a lying state before and after a purely falling motion. Therefore, in the preprocessing stage, we labeled timepoints when falls started and ended through comparing to videos that recorded the experiment. Data consisted of 70% train dataset for learning and 30% test dataset for validation. The LSTM model, which has strengths in analyzing time series data, was trained to learn features of falls, and we verified its performance of classifying falls versus non-falls. As a result, fall classification performance had an accuracy of 99% indicating that this approach was effective as a model for fall detection.

5.2 Limitations

There are several limitations in applying the contents of this study as an actual fall detection system. First, in addition to the movements defined as falls and ADL in this experiment, various movements that may occur in real life should be included. In the case of fall movements, slipping or tripping over objects frequently happen in reality. Thus, it is necessary to additionally conduct experiments and collect data on ADL movements as well as various fall movements.

5.3 Future Works

To implement falls and non-falls more accurately, we plan to further define and experiment various movements that may occur in addition to the 12 ADL movements and the four fall movements. It is important to accumulate abundant data for the model to learn, focusing on falls and non-fall movements that occur frequently in life.

Additionally, to develop the results of this study into a service that can be applied in real life, a series of systems should be developed which consists of sensing, communication, a fall detection process, and a user application program.

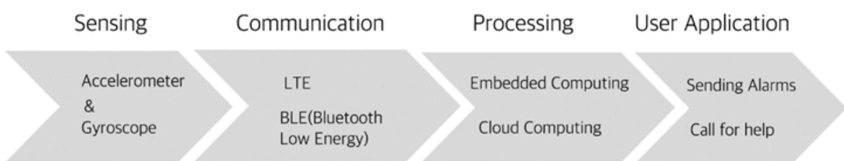


Figure 5.1: Model of the fall detection system.

When a fall occurs indoors, BLE [3] communication can be used to connect the sensed data to the beacon and determine the location of the fall.

[3]BTE (Bluetooth Low Energy) is a short-range communication method that consumes less power and is applied to smart bands and smart phones.

When a fall occurs outdoors, data can be transmitted using LTE[4] communication network. The calculation of fall occurrence through acceleration and angular velocity information can be analyzed through embedded systems of the wearable device itself, or through a cloud server for more accurate calculations. Finally, creating a user-centered scenario is considered the most important factor for commercializing products and services based on this study. In fact, the Samsung Electronic's Galaxy Watch and Apple's Apple Watch, which are currently sold with fall detection functions, detect fall accidents based on data collected through accelerometers and gyro sensors and on algorithms within smart watches. These products consist of scenarios in which the user can directly enter whether a fall has occurred for about one minute when detected as a fall, as a way to prevent false detection in order to provide a satisfactory fall detection function to the user. To improve usability, the quality of service can be improved with appropriate scenarios for the user, as well as enhancing the accuracy of the fall detection model.

One of the weaknesses of wearable devices is batteries. The wearable device used for fall detection is operated on battery in an environment where constant power supply is not possible. Therefore, the speed at which the battery of the wearable device is consumed must be considered. The factors that greatly exhaust batteries are the communication process for transmitting data and the sampling rate for collecting data. Thus, it is necessary to optimize the appropriate sampling cycle of sensors for fall detection and the appropriate cycle in the process of data transmission to the cloud server for

[4]LTE (Long Term Evolution) is a 4G broadband wireless communication technology developed by the 3GPP consortium. Standards such as Cat.M1 have been commercialized as IoT dedicated networks focused on small-scale data transmission.

fall verification. In addition, it is important to develop models based on GRU model or machine learning which lightens the LSTM to allow fall detection algorithm to be applied embedded in the device.

Bibliography

[1] WHO, "Falls." Available online at: https://www.who.int/news-room/fact-sheets/detail/falls, 2018.

[2] 정필성 and 조양현, "사물인터넷 기반의 낙상 감지 시스템 [fall detection system based internet of things]," *한국정보통신학회논문지*, vol. 19, no. 11, pp. 2546–2553, 2015.

[3] P. F. C. El-Bendary N., Tan Q. and L. A., "Fall detection and prevention for the elderly: a review of trends and challenges." DOI: https://doi.org/10.21307/ijssis-2017-588.

[4] 정낙수 and 최규환, "노인낙상의 원인과 예방 [cause and prevention of falling in the elderly]," *한국전문물리치료학회지 = Physical Therapy Korea*, vol. 8, no. 3, pp. 107–117, 2001.

[5] T. R, "Falls in older per sons : Prevention and manangement (2nd ed.)," *Baltimore, Health Professions Press*, 1998.

[6] 박영혜, "지역사회 재가노인의 낙상 위험요인 사정 [fall risk assessment in the community-dwelling elderly]," *노인간호학회지*, vol. 6, no. 2, pp. 170–178, 2004.

[7] Z. H. L. Y. C. C. Lin, C. W. and C. J. Kuo, "Compressed-domain fall incident detection for intelligent homecare," *The Journal of VLSI Signal Processing*, vol. 49, no. 3, pp. 393–408, 2007.

[8] 강윤규 and 원달수, "Posenet과 gru를 이용한 skeleton keypoints 기반 낙상 감지 [human skeleton keypoints based fall detection using gru]," *한국산학기술학회 논문지*, vol. 22, no. 2, pp. 127–133, 2021.

[9] G. E. Lee and J. W. Lee, "Comparison study of web application development environments in smartphone]," *Journal of KOCON*, vol. 10, no. 12, pp. 155–163, 2010.

[10] A. C. M¨uller and S. Guido in *Introduction to machine learning with Python: a guide for data scientists*, O'Reilly Media, Inc., 2016.

[11] 신수연 and 차주헌, "다중모드 센서와 lstm 기반의 딥 러닝을 이용한 인간의 행동인식 시스템 [human activity recognition system using multimodal sensor and deep learning based on lstm]," *Journal of KO-CON*, vol. 42, no. 2, pp. 111–121, 2018.

[12] J. S.-B. L. J.-H. Mun, S.-E. and J.-S. Lee, "기계학습 및 딥러닝 기술동향 [machine learning and deep learning technology trends]," *Information and Communications Magazine*, vol. 33, no. 10, pp. 49–56, 2016.

[13] T. R. Mauldin, "Smartfall: A smartwatch based fall detection system using deep learning," *Sensors*, vol. 18, no. 10, p. 3363, Oct., 2018.

[14] C. Chatzaki in *Human Daily Activity and Fall Recognition Using a Smartphone's Acceleration Sensor*, pp. 100–118, in Proc. Inf. and Commun. Tech. for Ageing Well and e-Health(ICT4AWE 2016), July, 2017.

[15] 남정우 and 김진헌, "가속도 센서 데이터 기반의 행동 인식 모델 성능 향상 기법 [improving performance of human action recognition on accelerometer data]," *전기전자학회논문지*, vol. 24, no. 2, pp. 152–157, 2020.

국문초록

딥러닝 기반 가속도 및 자이로 센서 데이터 활용 낙상감지 방법

이해성
서울대학교 공학전문대학원
응용공학과 응용공학전공

전 세계가 초고령화 사회로 진입함에 따라 노인 낙상 사고가 크게 증가하고 있다. 이러한 낙상 사고는 제때 감지되지 않을 경우 최악의 경우 사망까지 이르는 심각한 결과를 초래한다. 따라서 낙상이 발생했을 때, 즉시 낙상을 감지할 수 있는 시스템 구축이 필요하다. 낙상을 발견하기 위한 다양한 방법 중에서 착용이 쉽고 실내외에서 적용이 가능한 웨어러블(Wearable) 장치의 형태를 고안한다. 본 연구는 웨어러블 기반 가속도 센서와 자이로 센서를 활용하여 착용자의 움직임을 측정하고, 가속도 및 각속도 값을 분석하여 낙상 발생 여부를 분류하는 모델을 개발하고자 한다. 데이터를 획득하기 위하여 피실험자에게 웨어러블 장치를 착용한 상태로 12가지 일상생활동작과 4가지 낙상동작을 반복적으로 실시하는 실험을 수행하였다. 일상생활동작은 앉기, 서있기, 걷기 등이 있고, 낙상동작은 앞으로 넘어지는 동작, 뒤로 넘어지는 동작 등으로 구성된 데이터를 확보하였다. 낙상과 비낙상 여부를 검출하기 위하여 딥러닝 알고리즘 모델 중 순환 신경망(Recurrent Neural Network, RNN) 계열의 LSTM을 활용한다. 낙상

여부를 판단하는 LSTM 모델의 입력 값에 적용되는 데이터 전처리 및 미세조정(Fine-Tuning) 방법을 통해서 모델을 고도화 하였다. 실험 환경에서 모델의 낙상감지 정확도(Accuracy)는 99.91%로 심층학습 관점에서 낙상 검출의 타당성을 판단하고자 한다.

주요어 : 낙상; 6축 센서; 딥러닝; 시계열 데이터; LSTM.
학번 : 2020-20356