Fall Detection Based on Accelerometer and Gyroscope using Back Propagation

Adlian Jefiza[1], Eko Pramunanto [1 2], Hanny Boedinoegroho[1 2] , Mauridhy Heri Purnomo [1 2]
[1]Department of Electrical Engineering, [2]Department of Computer Engineering
Institut Teknologi Sepuluh Nopember, Surabaya Indonesia 60111
Email: adlian15@mhs.ee.its.ac.id , { ekopram, hanny, hery}@ee.its.ac.id

***Abstract* - Falling is an external aspect that can lead to death for the elderly. With so many activities they can do will increase the likelihood of falling. A fall detection device is designed to minimize post-fall risk. An MPU6050 sensor with 3 axis accelerometer and 3 gyroscope axis is used to detect the activities of the elderly. This research is expected to recognize the falling forward movement, falling aside, falling backward, sitting, sleeping, squatting, upstairs, down stairs and praying. The total data in the test is 80 data per movement of 16 participants. Backpropagation method is used for motion recognition. The recognition of this movement is based on 10 input variables from the accelerometer sensor data and gyroscope sensor. The result of this study, the error value calculated by using the formula Sum Square Error of all movements, is 0.1818 with ROC accuracy of 98.182%**

***Keywords* – Fall Detection, Accelerometer, Gyroscope, Backpropagation**

I. INTRODUCTION

Fall is a serious problem that is often experienced by the elderly in performing daily activities. The older the person the higher the percentage of falling possibility. The frequency of falling can make dangerous effect for the elderly, because the condition of their body is weak and susceptible. Many factors were originally considered as possible risk factors for falls based on a review of currently available literature. Reviews of these factors include age, number of chronic diseases, body composition, muscle strength, functional mobility and performance measures related to balance function [1]. Therefore it takes control in real-time by the family to the elderly.

Care for the elderly can be done manually or automatically with the help of electronic devices. The manual care is usually done by families of the elderly. It takes more time and effort which requires a family or nurse to be kept alert for 24 hours to supervise. This method is considered less efficient because the care taker also should do many things in their daily life. Therefore, electronic monitoring based on electronic devices can help monitor accurately.

The design of this surveillance tool has been developed by using sensors, cameras, and smartphones. Based on the previous research, the fall detector has used an accelerometer and depth sensor using a Kinect camera.

The design of this monitoring tool has been developed using the Accelerometer sensor [2], Gyroscope, ECG, camera, microphone and more. From the previous research, the fall detector uses an accelerometer and depth sensor using a Kinect camera. The results were not getting the maximum accuracy[3].

Some previous studies used several algorithms or methods that fit the sensor. Simple algorithms typically use a threshold value[4]. The use of this threshold algorithm is usually modified by adding the upper threshold or lower threshold. In addition to using the threshold algorithms, the use of data mining is also widely applied, eg K-Nears Neighboards (KNN), Support Vector Machine (SVM). However, the accuracy of the fall detector has not been satisfactory and there are still many weaknesses.

In this paper we are using accelerometer and gyroscope sensors. Both of these sensors are embedded in the module mpu6050 [5]. Accelerometer and gyroscope sensor provide acceleration of body changes and body rotation speed. This research is into innovation in detecting falls by adding a feature extraction and variable input. And for the introduction of its activities, Backpropagation Neural Network (BPNN)is used..

Another renewal in this study is to test the prayer movement, where in the previous studies no one has tested the movement. The purpose of this study is to use Backpropagation which is expected to recognize the elderly activities and get a high accuracy in detecting the fall.

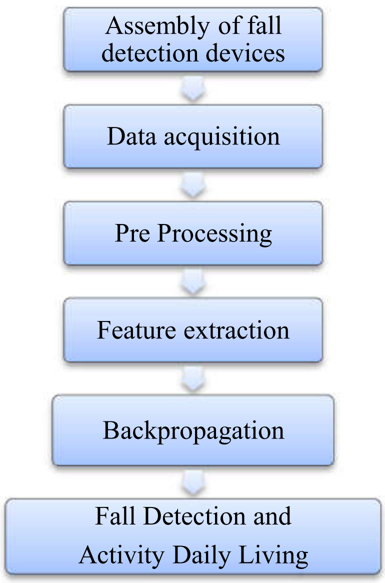

Fig. 1 The Procedure of Research

II. DATA AND RESEARCH METHOD

A. *The Procedure of Research*

There are several stages in this research. To illustrate the flow of research, it can be seen in Figure 1. The first stage is assembly of the detection falls device. The second stage is data acquisition by conducting experiment to participant in doing some fall movements. The third stage is the raw data that has been obtained on the data acquisition required preprocessing stage where the data is normalized. The last stage is classifying the kind of fall movement by using Back Propagation Neural Network method.

B. *Fall Detection Device*

For fall detection devices, the components used the MPU 6050 module which has 3 axes Accelerometer and 3 Gyroscope axes. The sensor is embedded in a microcontroller that uses Arduino Nano ATMEGA 328. The data is stored in the SD Card module. Recognition using the fall detection device can be seen in Figure 3. There are four components in the fall detection device that is MPU6050 sensor, microcontroller, storage media and sensor data processing.

C. *Acquisition of Data*

The data used was obtained from MPU 6050 sensor that has 3 axis accelerometer and 3 axis gyroscope. MPU 6050 sensor uses a scale of 2g, 4g, 8g and 16g for Accelerometer and scale of 250, 500, 1000, and 2000. In this study, the accelerometer used sensor of 16g scale and 250 for gyroscope sensors. The mpu6050 sensor is connected to Arduino Nano and is set at a frequency of 20 Hz.

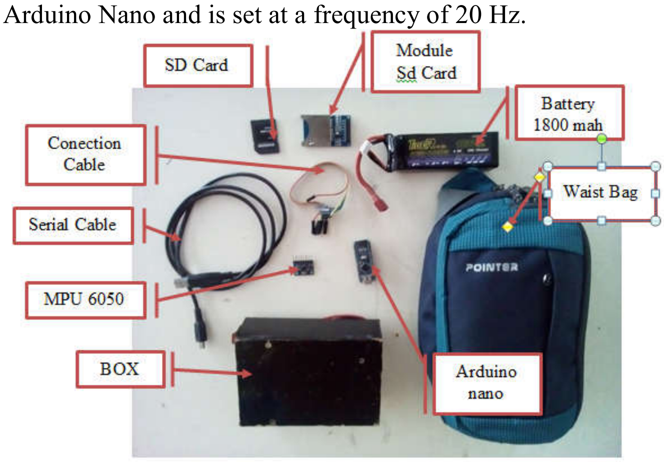

Fig 2 Component of The Fall Detection Devices

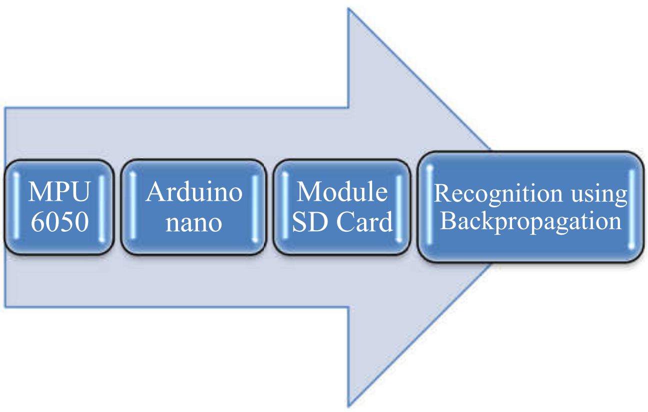

Fig 3. Fall detection device

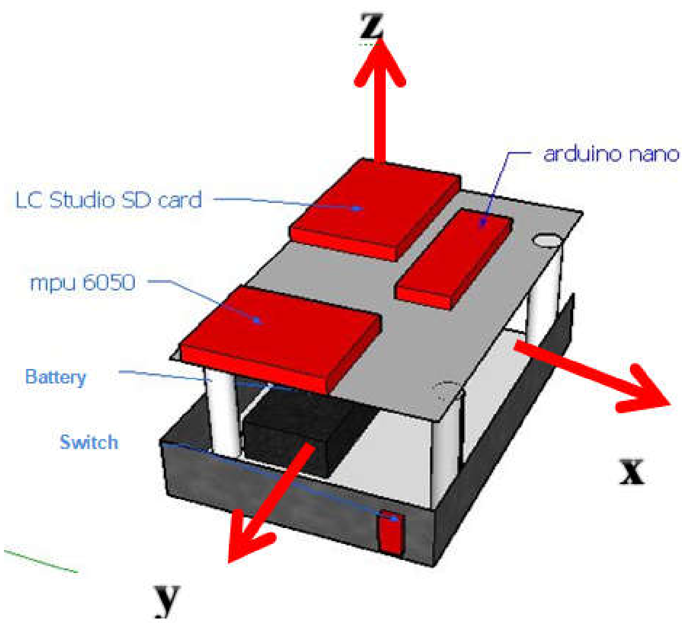

Fig 4 Desain of The Fall Detection Device

Table 1. Range of MPU6050[5]

Range Accelerometer	Scale	Range Gyroscope	Scale
2g	16,384	250 °/s	131
4g	8,192	500 °/s	65,5
8g	4,096	1000 °/s	32,8
16g	2,048	2000 °/s	16,4

The data was retrieved at least 10 seconds in a movement except for the prayer movement. The data was stored on the SD card module, and processed in the computer. The data obtained is still in the form of raw data, the data were processed according to the scale used. In this data, there were 5 movements. They were 9 fall movements (fall forward, fall beside and fall backward), sit, squat, up stairs, down stairs, sleep and pray.

The use of the prayer movement is very important because there is no research that uses prayer as a movement in trial. The total data obtained in the form of 240 data falling and 640 daily activity data with 3 movements in prayer. 3 movements in prayer are bow, movement from I'tidal to prostration and prostration.

Raw data MPU6050

The data obtained from the sensor mpu6050 accelerometer X, Y and Z axis were termed as (Ax), (Ay), and (Az).While gyroscope was denoted as (Gx), (Gy), and (Gz).

Based on the scale used, the raw data must be shared by using this formula:

$$A_{axis} = \frac{scale}{A\,raw_{axis}} \qquad (1)$$

A_{axis} is the axis which will be calculated for example Ax, Ay and Az. The scale used is 16 g and adjusted to the table, while $a\ raw_{axis}$ is the raw value of the x, y, and z accelerometer. As for gyroscope also uses the formula as follows :

$$G_{axis} = \frac{scale}{G\,raw_{axis}} \qquad (2)$$

G_{axis} is the axis which will be calculated such as Gx, Gy and Gz. The scale used is 250 and adapted to the table, while the G raw_{axis} is the raw value of the x, y, and z Gyroscope.

After the data is calculated based on the scale, the magnitude value of the xyz axis data on the accelerometer is calculated by using the following formula:

$$ATt = \sqrt{aX^2 + aY^2} + aZ^2 \quad (3)$$

Beside the magnitude value, the slope of the accelerometer is also used as an additional feature or often called pitch roll and yaw. The following formula is used to calculate the slope.

$$arx = \left(\frac{180}{\pi}\right) * \text{atan}\left(\frac{ax}{\sqrt{ay^2+az^2}}\right) \quad (4)$$

$$ary = \left(\frac{180}{\pi}\right) * \text{atan}\left(\frac{ay}{\sqrt{ax^2+az^2}}\right) \quad (5)$$

$$arz = \left(\frac{180}{\pi}\right) * \text{atan}\left(\frac{\sqrt{ay^2+ax^2}}{az}\right) \quad (6)$$

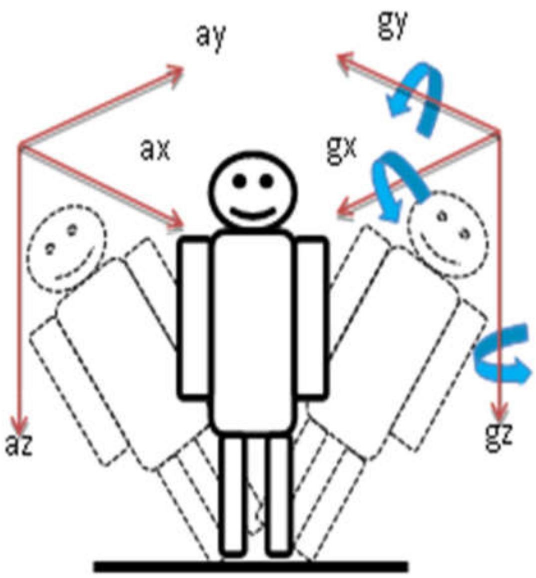

Fig 5. Accelerometer and gyroscope for Fall detection

Fig 6. Installation of fall detection device on the body

D. ***Pre Processing***

Within 10 minutes of data retrieval, usually there is a minimum of 140 per each of the x, y and z axes. So the last 100 data taken as raw data will be normalized. The next step is to change the x, y, and z axes of the sensors on a scale of 0-1. It aims to normalize the data by using the formula as follows:

$$normalized(x) = \frac{x - minValue}{maxValue - minValue} \quad (7)$$

Normalized data will be extracted by features based on average values, maximum values and minimum values.

E. ***Back Propagation Neural Network***

In principle, BPNN is like a human brain which can learn and save. BPNN model is made up of many layers of nodes and is designated by the node characteristics, network interconnection geometry, and the learning rules (transfer functions) [2]

These learning rules feed back into the model to change the weight of nodes between layers in order to decrease errors between predicted and measured values [1]. The structure of BPNN in this research consists of 3 layers, input, hidden and output layers as shown. Each layer consists of many

neurons. Let a, b, and c become the input, hidden and output neurons. In general, node a is the number of inputs, in this study means the axis of the accelerometer and gyroscope, c is the expected output pattern, here means falling or not. Overall, the work process of BPNN is located at node b. If the value of b is too small. Some train data as input data is provided on the network. The network will calculate output value, if there is error (between the desired value and the value obtained) then the weight in the network will be updated.

To start the motion recognition using backpropagation, all input data and output data must be transformed in the range 0-1. Totalize the result of multiplication of inputs with their respective weights using formula 8 as follows:

$$vj(p) = \sum_{i=1}^{r} xi(p).wi\,j(p) \quad (8)$$

The total value of v is derived from the sum of the multiplication of each feature by weight. The value of v is in activation by using the binary sigmoid activation function to get the value on the hidden layer.

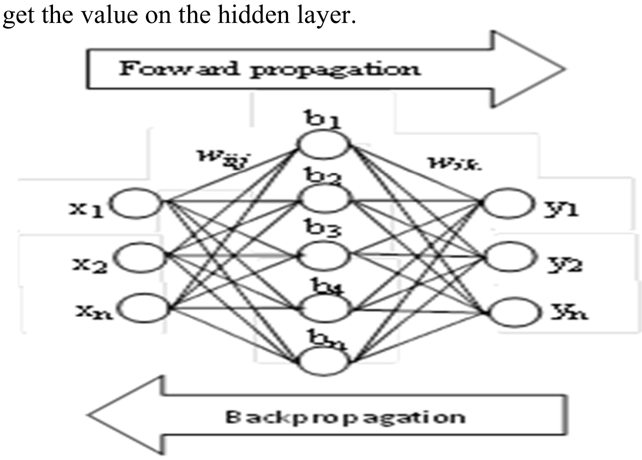

Fig 7 BackPropagation Neural Network

$$y_k(1) = \frac{1}{1 + e^{-vj(p)}} \quad (9)$$

Yk (1) represents the value in the first neuron. Next stage is calculated the value of each neuron. Each neuron is functionalized using a binary sigmoid. The value on the hidden neuron becomes the input. The value of each hidden neuron is multiplied by the hidden-output weight.

$$vj(p) = \sum_{i=1}^{r} xk(p).wjk(p) \tag{10}$$

The value obtained from the equation 10 is reactivated according to equation 11

$$y_k(1) \quad = \frac{1}{1 + e^{-vj(p)}} \tag{11}$$

From the calculation, there is still difference between the values obtained with the expected value. Error calculation can be calculated by formula 12-13.

$$e_k(p) \quad = y_{dk}(p) - y_k(p) \tag{12}$$

$$\delta_k(p) = y_k(p) \text{ x } [1 - y_k(p)] \text{ x } e_k(p) \tag{13}$$

Weight correction with its learning rate 0.1:

$$\Delta w_{jk}(p) = n \text{ x } y_j(p) \text{ x } \delta_k(p) \tag{14}$$

III. EXPERIMENT

A. Calibration

The first step after assembly of the tool is calibration of MPU6050 sensor. This calibration aims to get the value of zeros G offset. The calibration is done by placing the sensor in horizontal position, and run the calibration program so that it gets the value 0 on the x axis 0 on the Y axis and 1 on the Z axis of the Accelerometer sensor. This calibration value will be incorporated into the program on the microcontroller

B. Motion Capture System

The motion capture uses MPU6050 sensor that is connected to Arduino Nano microcontroller and the data is stored on micro SD. For the first stage, the device tested is done by searching for the normal value of the mpu6050 sensor in certain positions such as sitting, standing, sleeping, praying and other movements. This data collection aims as a reference for output value. The position of the device is placed in the waist bag horizontally, the x-axis facing the left of the body, the y-axis upwards and the z-axis leads to the body, as shown in the picture. The results are obtained on the fall detection device in validation with the accelerometer sensor value on the smartphone. The results of the value of each axis to the movement of the test can be seen in Table 2.

C. The Human Fall Detection

In this study, the movement is categorized into two groups, namely falling motion and daily activities. For falling movement, there are 3 types of movement that is falling forward, falling aside and falling backward. The daily activities movement includes sitting, squatting, sleeping, upstairs, down stairs and praying.

All movements were repeated for 5 times movement in 7 seconds. Each movement was counted as one data. It means that for 1 person there are 15 fall movements and 30 movements of everyday activities.

From the accelerometer and gyroscope data obtained from falling motion and daily activities, a maximum value is sought as a movement pattern recognition feature. For the introduction of this movement can be processed using Backpropagation Neural Network

Fig 7. The Experiment of Movements Falling Forward

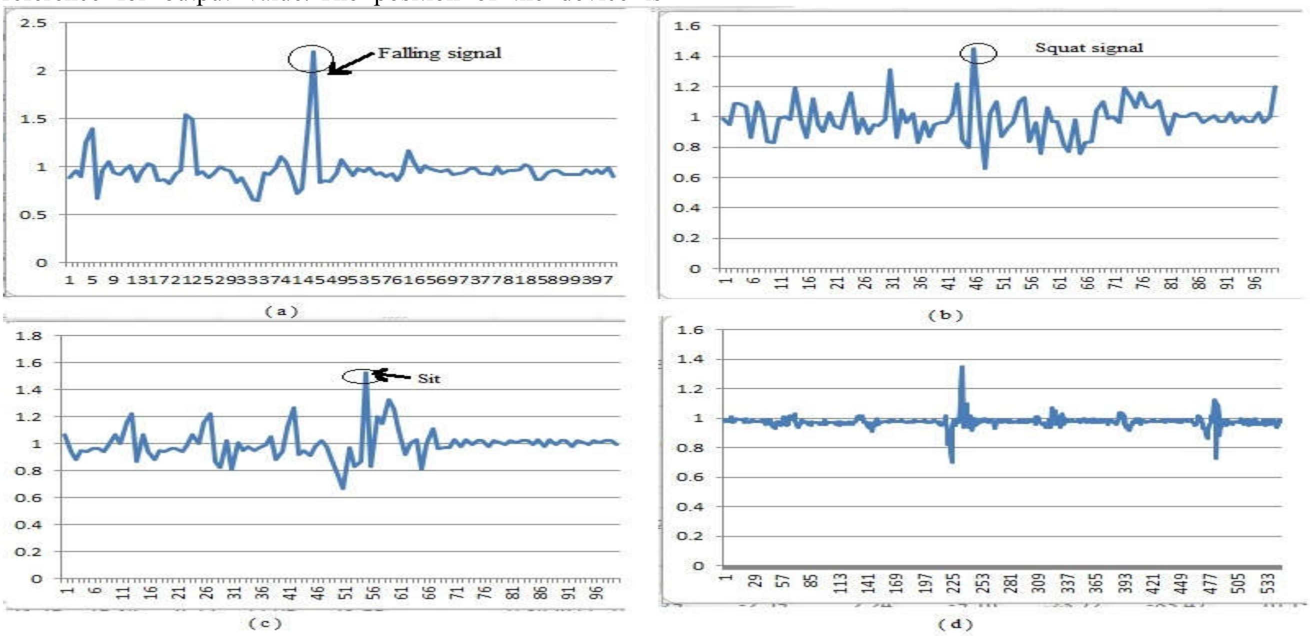

Fig 8 Feature of alpha value in each movement
a) Falling forward; (b) Squat; (c) Sit; (d) Prayer

Table 2. Feature Extraction Based On Average Values, Maximum Values And Minimum Value

	Ax	Ay	Az	Alpha	Gx	Gy	Gz	Pitch	Roll	Yaw
Falling to the side	0.56	0.68	0.23	0.72	0.59	0.42	0.56	0.58	0.55	0.10
Falling backwards	0.55	0.96	0.75	0.76	0.10	0.57	0.51	0.45	0.75	0.66
Falling forward	0.62	0.80	0.22	0.71	0.71	0.50	0.40	0.56	0.66	0.33
Sit	0.44	0.52	0.74	0.37	0.65	0.49	0.46	0.36	0.80	0.86
Sleeping	0.31	0.46	0.74	0.31	0.48	0.52	0.35	0.39	0.54	0.64
Squat	0.53	0.44	0.49	0.38	0.54	0.53	0.50	0.44	0.77	0.09
Up stairs	0.61	0.78	0.54	0.68	0.41	0.55	0.59	0.57	0.63	0.53
Down stairs	0.42	0.77	0.29	0.62	0.50	0.45	0.51	0.36	0.78	0.09
Bow in prayer	0.66	0.33	0.34	0.43	0.49	0.56	0.49	0.67	0.40	0.40
I'tidal	0.48	0.75	0.75	0.59	0.43	0.56	0.48	0.48	0.94	0.60
Prostration in prayer	0.60	0.20	0.40	0.43	0.51	0.56	0.50	0.61	0.24	0.30

IV. RESULT

This section will explain the results obtained by retrieval of data, process the data and see the value of accuracy. For raw data of falling and other activities can be observed based on picture 6. There is similarity of alpha value in every movement. Data is processed using backpropagation neural network with 11 input variables that have been separated based on its features, 10 hidden layers and one output. The amount of data used is 660 220 training data and testing data obtained from 11 movements made up of three movements of fall, 5 Activities of Daily Living and 3 prayer movements. The results of this study can be seen in table 2. From the recognition value of this movement to calculated error value of all movements, the SSE formula is used. The result of error value is 0.1818. ROC is used to calculate accuracy, precision, sensitivity and specification. To obtain the magnitude of accuracy, ROC is calculated using True Positive (TP), False Positive (FP), True Negative (TN), False Negative (FN) values. Motion recognition is done at 30,000 iterations. For each tested data is shown in Table 4.

Table 3. Accuracy in Iteration 30.000

Movement	Data	Recognize	Accuracy
Falling to the side	20	20	100%
Falling backwards	20	19	100%
Falling forward	20	20	90%
Sit	20	18	100%
Sleeping	20	20	100%
Squat	20	20	100%
Up stairs	20	20	90%
Down stairs	20	19	85%
Bow in prayer	20	20	100%
I'tidal	20	20	90%
Prostration in prayer	20	20	100%

In Table 4. there are three tests, ie at 10,000, 20,000 and 30,000 iterations. This test is performed to find the best accuracy when viewed from the number of iterations. The third result of the test, iteration 30,000 obtained the best accuracy

Table 4. Motion Recognition Based on Iteration

	TP	FP	TN	FN
Iteration 10.000	55	5	155	5
Iteration 20.000	58	2	157	3
Iteration 30.000	59	1	157	3

From the data in table 4, the value of accuracy, precision, specification and sensitivity can be calculated, and the calculation results can be seen in table 5.

Table 5 The Result of Motion Recognition

	Precision	Specification	Accuracy	Sensitivity
Iteration 10.000	91.667 %	96.875 %	95.455 %	91.667 %
Iteration 20.000	96.667 %	98.742 %	97.727 %	95.082 %
Iteration 30.000	98.333 %	99.367 %	98.182 %	95.161 %

V. CONCLUSION

The results of this study have got a high accuracy. Of the three tests with different iterations, the best value was obtained at 30,000 iterations. The value of error that can be calculated with formula SSE is 0.1818 with an accuracy of 98.182%, precision of 98.33%, sensitivity of 95.161%, and specification of 99.367%. The way of data collection greatly affects the results of this study.

This research can be developed in hardware or method. Expected at the development stage can obtain more accurate results with more effective and efficient hardware.

BIBLIOGRAPHY

[1] A. Ozcan, H. Donat, N. Gelecek, M. Ozdirenc, and D. Karadibak, “The relationship between risk factors for falling and the quality of life in older adults,” *BMC Public Health*, vol. 5, no. 1, Dec. 2005.

[2] P. Siirtola and J. Röning, “Recognizing human activities user-independently on smartphones based on accelerometer data,” *IJIMAI*, vol. 1, no. 5, pp. 38–45, 2012.

[3] B. Kwolek and M. Kepski, “Improving fall detection by the use of depth sensor and accelerometer,” *Neurocomputing*, vol. 168, pp. 637–645, Nov. 2015.

[4] A. Kurniawan, A. R. Hermawan, and I. K. E. Purnama, “A wearable device for fall detection elderly people using tri dimensional accelerometer,” in *Intelligent Technology and Its Applications (ISITIA), 2016 International Seminar on*, 2016, pp. 671–674.

[5] F. Vora, S. Kanojia, M. Sayad, and F. Sayyad, “GESTURE CONTROLLED POWERPOINT.”

[6] R. Hecht-Nielsen and others, “Theory of the backpropagation neural network.,” *Neural Netw.*, vol. 1, no. Supplement-1, pp. 445–448, 1988.