TEST Engineering & Management
November-December 2019
ISSN: 0193-4120 Page No. 2404 – 2409

Discrimination of Sleep Posture Using CNN in Smart Bed

Tae-Hwan Kim[1], Ki-Young Lee[2], Youn-Sik Hong[1]
[1]Department of Computer Science and Engineering, Incheon National University, Incheon, {tomxoghks789, yshong@inu.ac.kr}
[2] Department of Information and Telecommunication Engineering, Incheon National University, Incheon, kylee@inu.ac.kr

Article Info
Volume 81
Page Number: 2404 – 2409
Publication Issue:
November-December 2019

Article History
Article Received: *5 March 2019*
Revised: *18 May 2019*
Accepted: *24 September 2019*
Publication: *12 December 2019*

Abstract
Sleep posture is one of the important indicators to measure the health status of a person. The purpose of this paper is to know which posture a sleeping person took while sleeping and to know which body part of the sleep posture produced the strongest pressure. To do this, we have implemented a smart bed with a set of FSR sensors arranged in a grid structure. The pressure intensity of each FSR sensor is converted into the corresponding gray image. Then the convolutional neural network composed of eight layers was applied to the 50 samples to discriminate one of the three lying postures. The results showed 94.42% accuracy, which is more than 7% higher than the typical method based on the distribution of pressure intensity.

Keywords: *CNN, FSR sensor, Lying posture, Smart bed, WiFi*

1. INTRODUCTION

Sleeping posture is one of the most important habits because a person sleeps in bed in one third of his/her life. It also affects to his/her daily routine. In addition, sleep posture is one of the important indicators to measure the health status of a person. For example, a prone position should turn your face sideways to breathe, which can cause pain in the back, giving the neck a burden. Therefore, there is a need for a technique that continuously monitors one's sleeping posture.

In this paper, we propose a method of recognizing sleeping posture by utilizing a smart bed with a set of FSR (force sensing resistor) sensors arranged in a grid structure [1]. The posture of a person lying on the smart bed varies widely from normal (or upright) posture to bent posture, prone posture, and so on. The purpose of this paper is to know which posture a sleeping person took while sleeping and to know which body part of the sleep posture produced the strongest pressure. In other words, in the case of an ordinary person, the change in sleep posture will be tracked. However, a patient who is unable to move alone will be monitored to track the pressure applied to his/her body parts in the lying posture.

In our previous works [1], we already proposed a method of monitoring lying posture through the analysis of pressure distribution. In this paper, the pressure intensity between 0~1,023 is converted into a grayscale image between 0~255. The smart bed consists of 16x8 (=128) FSR sensors. The pressure intensity sensed by each sensor is converted to a grayscale image. Thus, the image size for one lying posture will be 16x8, the same as size of the smart bed. Due to the very small size of the image, the training time is very short. A strong pressure appears as a bright color (i.e., white color), whereas a weak pressure

November-December 2019
ISSN: 0193-4120 Page No. 2404 – 2409

appears as a dark color (i.e., black color). Then we want to discriminate the lying posture by applying the convolutional neural network (CNN) technique with 8 layers, which has recently been in the spotlight in image classification such as hand written digits.

2. THE PROCESS FLOW OF THE SMART BED SYSTEM

In this paper, the FSR 406 sensor [2] is used to measure pressure. As shown in Figure 1(left), it has a square shape and the size of 43.69mm x 43.69mm. When the pressure is applied to it, it returns a resistance value that is inversely proportional to the pressure as shown in the Figure 1 (right). The pressure range that can be measured by the FSR 406 sensor is from 0.1N to 10N. The pressure value sensed by the FSR sensor is stored as a digital value between 0 (0N) and 1,023 (10N).

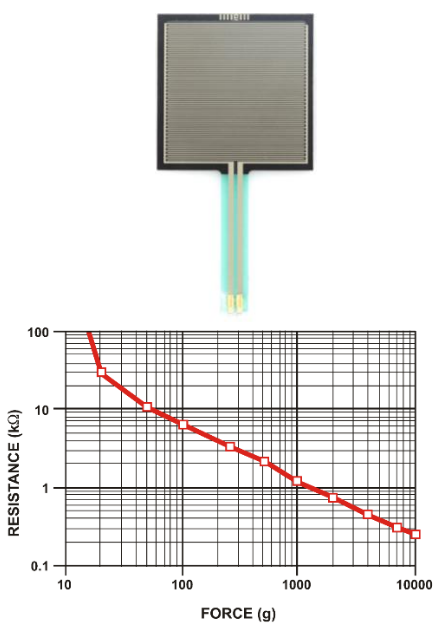

Figure 1: The FSR-406 sensor and its characteristic curve[2]

The total size of the smart bed is 189.9cm x 90.9cm because of the medical bed standard in Korea. The smart bed consists of 16x8 (=128) FSR sensors. We placed the sensors at intervals of 8 cm in width and length for efficient measurement. The smart bed is divided into four distinct subsections, each consisting of a total of 32(4 rows x 8 columns) FSR sensors.

We implemented Smart bed with Arduino Mega because of its characteristic. Unlike the commonly used Arduino Uno (it has 6 analog pins), Arduino Mega has more analog pins. it has 16 analog pins, so we can connect more FSR sensors. By implement with fewer controllers, we reduce the amount of control and simplified the data transfer program. The implemented Smart bed is shown in Figure 2.

Subsections are managed by four independent controllers implemented with Arduino Mega boards. These controllers are also hardwired to the concentrator implemented with Raspberry pi 3 board. It serves as a buffer to temporarily store the collected sensed data, and wirelessly transmits them to the centralized server at regular intervals via wireless LAN (IEEE 802.11b). The server uses MySQL as DBMS [3].

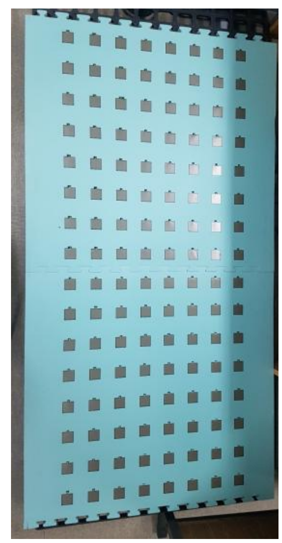

Figure 2: The implemented Smart bed

An artificial intelligence technique is applied to the dataset accumulated in the server to determine the lying posture. Figure 3 shows the process flow of the smart bed system.

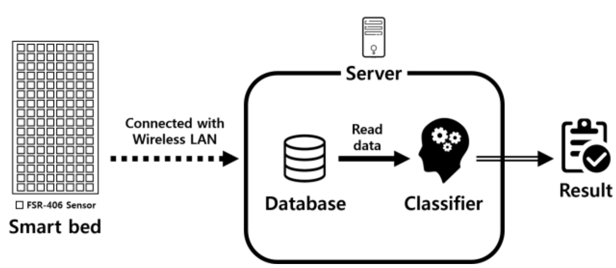

Figure 3: The process flow of the smart bed system

November-December 2019
ISSN: 0193-4120 Page No. 2404 – 2409

3. THE LABELED DATASET GENERATION FOR SUPERVISED LEARNING

For data generation, the experimenter lay on the smart bed shown as Figure 4, and the data of 16x8 size for each posture was sampled at regular intervals (every 1 minute). However, in order to change the lying posture, we asked the experimenter to change to another posture at each sampling. This work allowed us to measure the diversity of postures that differed slightly from person to person. This work improved accuracy. In order to determine the lying posture using the supervised learning method, the sampled training data should be labeled.

To this end, a separate program as shown in Figure 4 was implemented with C# and a data set was built by labeling each sample data. Through this program, we can preprocess the data. if there is outlier make value by 0 by click the cell which has outlier value. Also we can read the data sent from the Smart bed and stored in the database. By using this program to label the data. After labeled data is saved in CSV format for later training.

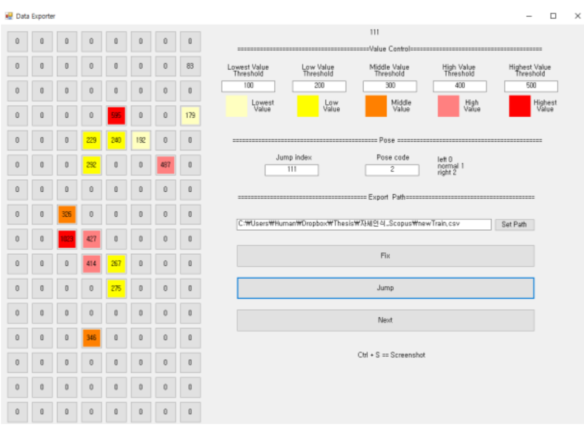

Figure 4: The snapshot of data labeling program

The data labeling program shown in Figure 4 is used to attach a label based on the result determined by a person after looking at the loaded data. If the head is facing left, the label is set to 0. If a person lies down in an upright posture (regular posture), it is set to 1. It is set to 2 if the head is facing right. The Figure 5 is an image of an experimenter measuring data.

Figure 5: The example of data measurement

The output with labels is saved as CSV format. The 55 data were generated for each posture and thus created a data set consisting of total of 165 data. This dataset was used for both training and k-fold cross validation (k=3). Notice that the dataset was measured from three experimenters.

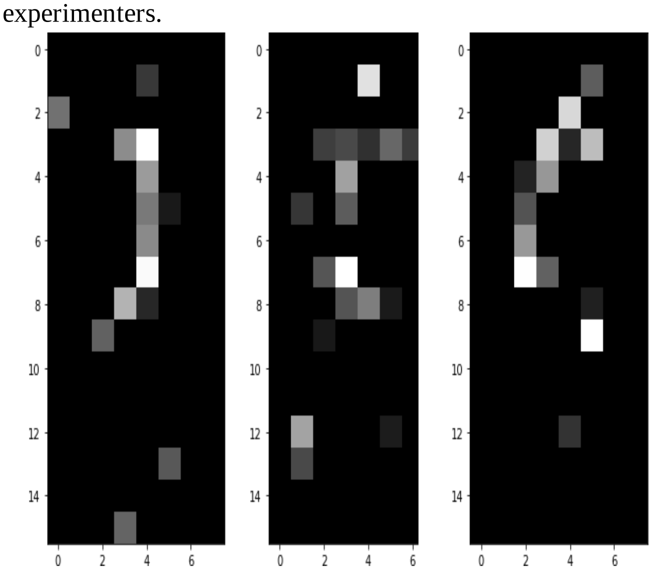

Figure 6: The visualized images of each posture (left, upright, and right posture)

The dataset built must be preprocessed before applying CNN. Since the smart bed system are configured with the analog sensors, there are cases where noise is included in the data. For example, if the pressure is measured even though the pressure is not actually applied, it is usually caused by noise. The preprocessing is needed to detect and adjust the abnormal value of the sensor (i.e., outlier). Figure 6 shows three images for the corresponding lying postures to be used for posture determination after preprocessed.

4. CNN ARCHITECTURE FOR CLASSIFYING SLEEP POSTURE

Instead of using the distribution of pressure intensity for classifying sleeping posture, it was converted into the image as shown in Figure 6. The image of each posture has its own characteristic, and this characteristic is sufficient to discriminate the posture. Thus, we will apply CNN (convolutional neural network) [4], a technique used to identify handwritten digit [5], to classify sleep posture. In this study, the CNN layers were constructed as shown in Figure 7.

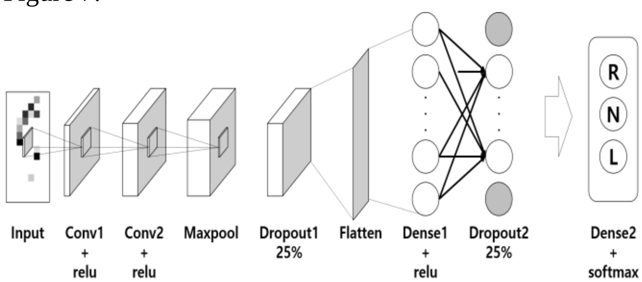

Figure 7: The CNN architecture with 8 layers

Notice that 70% of data sets are used as the training set, and the remaining 30% are used as the validation set. During the data set configuration, the data is selected randomly and not duplicated. When the training set is completed, the data is converted into four-dimensional data. The transformed data is passed through two convolution layers to extract the features. In both layers, the activation function uses the relu function. MaxPooling is performed to extract the key values from the output and to produce a small image.

With this work, minor changes do not affect the results. It then goes through a dropout that does not use 25% of nodes to avoid overfitting. The result will be converted to one-dimensional data through Flatten operation for a full connection. After another dropout through the hidden layer, the result is output through the Softmax function. We used Softmax as an activation function. Softmax is a function that is widely used in classification problems such as discriminating hand written digits. When using this, we need to specify how many classes to classify by using parameters. in this study we will classify 3 postures, so we input 3 as parameter. Table 1 shows the parameter values at each layer.

Table 1: Parameters set up at each layer

Layer	Paramete r	Value
Conv1	Filter	32
Conv2	Filter	64
MaxPoolin g	Pool size	2
Dropout1	Rate	25
Flatten	-	-
Dense1	Activation	128
Dropout2	Rate	25
Dense2	Activation	Softmax

Table 2 shows the parameter values set for configuring the CNN model. We use Softmax and Cross-entropy which are frequently used as loss function. Adam is used as the optimizer, but all parameters such as learning rate were used as defaults.

Table 2: Model parameters

Paramete r	Value
Loss	Cross entropy
Optimizer	Adam
Epoch	300
Bach size	20

5. EXPERIMENTAL RESULTS

We installed Tensorflow in an Anaconda virtual environment and implemented the proposed method using Keras running at Tensorflow [6]. In addition, the detailed H/W specifications are shown in Table 3.

Table 3: The H/W specifications used in the experiment

The total of 50 experiments took 948.82 seconds, with an average of 18.97 seconds per sample. For performance of a single sample, as shown in Figure 8, the loss and validation loss were 0.058 and 0.1838, respectively and the accuracy in the validation set was 0.9399(i.e., 94%). The total of 50 experiments showed an accuracy of 94.91% on average, with the minimum accuracy of 89.99% and the maximum of 98.00%.

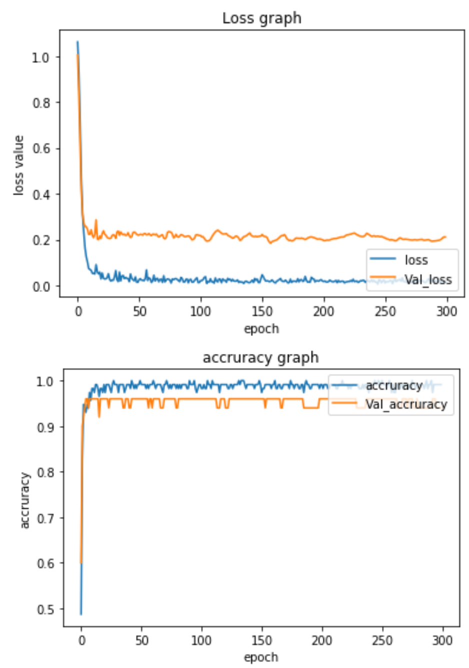

Figure 9: Loss graph(up) and accuracy graph(down)

Figure 9 shows some of the transformed image of the actual sleep posture. Notice that the word “Actual” in Figure 9 is the actual posture and the word “Pred” is the predicted result. The prediction of the five samples determined the correct posture except for the fourth posture from the left in Figure 9. The wrong predicted posture was lying to the left (0) in the actual posture, but it was discriminated as upright posture (1). It can be seen that the pixels lying in the straight line used to determine upright posture is overlapped with the posture lying to the left [7-8].

Type	Product name	Specification
CPU	Intel Core i7-9700K	3.60GHZ (Octa-core)
RAM	Samsung DDR4	16G PC4-21300 * 4
GPU	MSI Gaming X TRIO	GTX-2080
DISK	Samsung SSD 860 EVO	M.2 1TB
OS	Windows 10 Enterprise	-

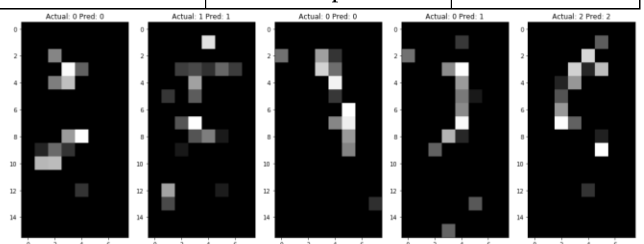

Figure 9: The prediction results

6. CONCLUSION

In this paper, to recognize the sleeping posture of a person lying on a smart bed, the pressure intensity is converted into the gray scale image, and then the 8 layer CNN architecture is applied. In the existing paper, the distribution of pressure position was used to determine the lying posture and the accuracy was 87%. Using our proposed CNN method, we have predicted more than 50 samples, showing 94.42% classification accuracy, in three distinct postures. We will increase the number of postures that can be categorized, and will continue to improve the performance so that it can be effectively applied to the sleeping posture recognition using the smart bed.

ACKNOWLEDGEMENT

This work is supported by Incheon National University Grant 2016.

REFERENCES

1. Tae-Hwan Kim, Soon-Ju Kwon, Hyun-Min Choi, and Youn-Sik Hong. **Determination of Lying Posture through Recognition of Multitier Body Parts**, *Wireless Communications and Mobile Computing*, vol. 2019, Article ID 9568584, 2019.

2. **FSR-408 and FSR-406**, *https://cdn-shop.adafruit.com/datasheets/ FSR400SeriesPD.pdf*.
3. S. Rautmare andD. M. Bhalerao. **MySQL and NoSQL database comparison for IoT application**, in *Proceedings of the 2016 IEEE International Conference on Advances in Computer Applications*, 2016 pp. 235–238.
4. N. Aloysius and M. Geetha. **A review on deep convolutional neural networks**, in *2017 International Conference on Communication and Signal Processing*, Chennai, 2017, pp. 0588-0592.
5. Y. Lecun, L. Bottou, Y. Bengio and P. Haffner. **Gradient-based learning applied to document recognition**, in *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278-2324, Nov. 1998.
6. **Tensorflow Module:tf.keras**, *https://www.tensorflow.org/api_docs/python/tf/keras* .
7. Jabarullah, N.H. (2019) Production of olefins from syngas over Al2O3 supported Ni and Cu nano-catalysts, Petroleum Science and Technology, 37 (4), 382 – 385.
8. Aastha, B., & Shazi, S. J. (2019). Corporate social responsibility practices in small and medium enterprises. Polish Journal of Management Studies, 19 (1), 9-20.