ISSN (Print) 1598-5385
ISSN (Online) 2233-6648
≪연구논문≫ Journal of the Korean Magnetics Society 31(4), 188-195 (2021) https://doi.org/10.4283/JKMS.2021.31.4.188

Human Activity Recognition Using Machine Learning Based Mobile Smart Phone Magnetometers

Chunghee Nam*
Department of Electrical and Electronic Engineering, Hannam University, Daejeon 34430, Korea

(Received 15 June 2021, Received in final form 28 June 2021, Accepted 29 June 2021)

As the performance of sensors embedded in mobile smart phones has improved, many studies using data collected from sensors are being conducted. In this study, using the data obtained from the 3-axis magnetic sensor mounted on the smartphone, a study on the recognition of four human activities was performed using machine learning. From the total data of the 3-axis magnetic sensor, the data was bundled into frames for 2 seconds, divided into several frames, and then supervised learning was carried out using it as an input to the convolutional neural network. The operation of the magnetic sensor depending on the direction was confirmed, and the human activity recognition for standing, sitting, walking, and jogging was verified.

Keywords : machine learning, convolution neural-network, human activity recognition, magnetometer

머신러닝 기반 모바일 스마트폰 자기센서를 이용한 인간행동인식

남충희*
한남대학교 전기전자공학과, 대전시 대덕구 한남로 70, 34430

(2021년 6월 15일 받음, 2021년 6월 28일 최종수정본 받음, 2021년 6월 29일 게재확정)

모바일 스마트폰에 내장된 센서의 성능이 향상되면서 센서에서 수집된 데이터를 이용한 연구가 많이 진행되고 있다. 본 연구에서는 스마트폰에 탑재된 3축 자기센서에서 얻어진 데이터를 이용하여 4가지 인간행동 인식에 대한 연구를 머신러닝을 이용하여 진행하였다. 3축 자기센서의 전체 데이터에서 2초간 데이터를 프레임으로 묶어서 여러 개의 프레임으로 나눈 다음, 합성곱 신경망의 입력(input)으로 사용하여 지도 학습을 진행하였다. 방향에 의존하는 자기센서의 작동을 확인하였으며, 서기(standing), 앉기(sitting), 걷기(walking), 뛰기(jogging) 에 대한 행동인식이 잘 이루어짐을 확인하였다.

주제어 : 머신러닝, 합성곱 신경망, 인간행동인식, 자기센서

I. 서 론

최근에, 컴퓨터 성능의 발달과 많은 데이터(빅데이타) 수집이 가능해지면서 인공지능을 포함하는 머신러닝의 활용분야는 시각(vision) 및, 음성 분야를 넘어서 다양한 영역에서 활발히 연구되며 응용되고 있다[1-3]. 각 종 데이터의 수집은 센서를 통해서 가능하며, 근래에는 사람이 소지하고 있는 각 종 모바일 IoT 센서를 통해서 지속적으로 데이터가 모아지고 여러 분야에서 활용되고 있다[4]. 스마트 폰은 단순한 통화장치 이상으로 활용되는 인간 생활의 필수품일 뿐 아니라 각종 센서가 탑재되어 있으며 이를 바탕으로 게임, 위치추적, 사람인식, 음성인식등에 활용되고 있다. 스마트폰을 활용하여 실시간으로 수집된 데이타를 기반으로 사용자의 행동인식, 위치추척 뿐 아니라 에너지 소모량 추정 또한 머신러닝 기술과 접목되어 활발하게 연구되고 있다[5,6].

스마트폰에 탑재된 센서를 활용한 행동인식 기법은 최근에 많은 연구가 이루어졌으며, 머신러닝 기초 교육법으로 활용될 수 있는 간단한 방법이다[7]. 스마트폰에 내장된 동적 감지 센서는 가속도센서, 자이로 센서, 자기장센서등이 있지만 주로 가속도 센서가 행동인식 기법에 주로 활용되었다[6,7].

자기장센서는 민감도가 우수하지만, 행동인식에 활용이 된

© The Korean Magnetics Society. All rights reserved.
*Corresponding author: Tel: +82-42-629-7472,
Fax: +82-42-629-7519, e-mail: chnam@hnu.ac.kr

경우가 많지 않다. 최근에 전자기유도방법을 적용하여 실험실에서 자체적으로 제작된 자기센서를 이용한 인간 행동인식에 대한 연구가 발표되면서 앞으로 자기센서의 활용도는 높을 것으로 예상된다[8-10]. 또한 자기센서의 특수성으로 인해서 지상에서 뿐 아니라 인공위성에서 방향감지 대한 연구는 오랫동안 이루어졌다. 하지만 행동인식에 사용이 제한된 이유는 다른 센서와 다르게 금속 물체등에 의해서 자기장 교란이 일어날 수 있기 때문이다. 하지만 이러한 문제는 다양한 방법으로 해결될 수 있음이 행동인식이 아닌 다른 연구분야에서 지속적으로 연구되어 왔다[11].

II. 데이터 수집 및 실험방법

위의 배경을 바탕으로, 본 연구에서는 금속등의 자기장 교란이 없는 환경에서 스마트폰에 내장된 3축 자기센서를 이용하여 행동인식을 머신러닝 기법으로 확인하였다. 행동인식 4가지인 걷기(walking), 달리기(jogging), 앉기(sitting), 서기(standing) 경우에 대해서 스마트폰에 내장된 3축 자기장 센서(Bx, By, Bz)의 데이터를 실시간 수집하였다. 10 Hz sampling rate 로 수집된 데이터는 텍스트파일(*.txt)로 실시간 저장되어 파이썬 프로그램을 통해서 머신러닝에 활용하였다. 머신러닝 작동환경과, 사용된 핸드폰 및 내장된 자기센서 사양, 파이썬 프로그램과 사용된 라이버러리(toolkit 또는 module)에 대한 내용은 Table I과 같다. MOUSER Electronics사의 HSCD-series의 geomagnetic sensor는 3축 홀센이며, 주요 specification은 Table I 아래에 나타내었다.

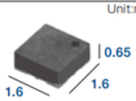
실시간 3축 자기장 센서에서 수집된 매 2초간 데이타를 한 프레임으로 설정하고, 전체 데이타에서 각 프레임은 1초 간격으로 나누어서 전체 프레임을 구성하였다. 각 프레임은 합성곱 신경망(Convolutional Neural Networks: CNN)의 input으로 입력되고 pooling 및 flatten layer와 fully-connected layer가 포함된 deep learning 방법으로 해석하였다[4,7].

Fig. 1은 사용된 스마트폰(iPhone 6S)과 스마트폰에 내장된 3축 자기장 센서의 방향 및 iOS 앱을 통해서 보여주는 임의의 실시간 데이터를 보여주고 있다[12]. 여기서 Bx는 스마트폰 가로방향, By는 세로방향, Bz는 지면을 통과하는 방향이다.

Table I. Basic details for the computer system, phone, embedded sensor, and python and its libraries.

OS, CPU , python library, sensor, phone	용도
64bit window 10 Home OS	운영체제
CPU processor : Intel(R) Core(TM) i7-10510U CPU @ 1.80GHz 2.30 GHz, RAM: 8 Gb	컴퓨터 specification
Python 3.7.9	coding
Tensorflow 2.4.0	딥러닝
pandas 1.2.0	Data frame
Numpy 1.19.2	math
Seaborn 0.11.1	plot
Scipy 1.5.2	통계
Matplotlib 3.3.2	plot
phone	iPhone 6S
Embedded sensor	Alps Electric HSCDTD007

Geomagnetic Sensors

Series	Type	Interface	Drive voltage	Operating magnetic field (resolution)
HSCD 	Electric compas (Surface mount)	I²C	Analog: 1.7 to 3.6V Digital: 1.65 to AVDD	Total with 3-axis: 7.2mT 1-axis: 2.4mT (Resolution: 0.15μT)

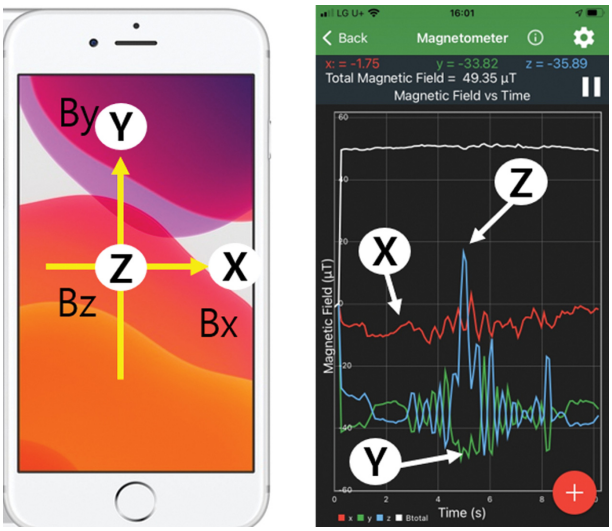


Fig. 1. (Color online) 3 axis of a magnetometer embeded in a mobile phone and examples of real-time data of Bx, By, and Bz.

III. 결과 및 토의

Fig. 2는 4가지의 행동인식 기법을 적용하기전에 3축 자기장 센서의 기본 데이터를 확인하기 위한 실험이다. 사용자가 고정된 자세에서 손으로 스마트폰을 잡고 x, y, z 방향으로 일정하게 움직이면서 100초 동안 데이터를 수집한 결과이다. 데이터 수집 시작과 마지막 과정에서 스마트폰 조작에 의한 이상치(outlier)를 제거하기 위해서 100초 동안의 1000 samples 가운데 90초 동안의 900 samples 데이터를 활용하여 나타내었다. Fig. 2(a)는 핸드폰을 x-방향으로 움직인 경우이므로 Bx의 데이터의 변화가 가장 큰 것을 알 수 있다. 마찬가지로 2(b)와 (c)는 각각 y 방향, z 방향으로 움직인 경우이므로 By, Bz의 신호크기가 큰 것을 알 수 있다. 이를 통해서 스마트폰에 내장된 3축 자기센서는 적절하게 작동함을 알 수 있다.

Fig. 3(a)는 실험자가 데이터 수집을 실시한 운동장 트랙과 방향을 보여준다. 운동장 긴 방향과 정북 방향이 일치함을 동일한 스마트폰의 자기센서인 나침반 기능을 사용하여 확인하였다. Fig. 3의 (c), (d), (e), (f)는 4가지 행동인식의 각 경우에 대한 Bx, By, Bz 신호를 5000 샘플링에 대한 결과를 보여준다. 자기센서를 이용한 4가지 행동인식에 대한 측정은 연속적인 5000 샘플링으로 그래프를 보여주며 이것은 시간으로 500초에 해당한다. Fig. 3(b)는 동일한 스마트폰의 g-force 센서(sampling rate = 100 Hz, 800초 동안 측정)를 통해서 측정한 신호이며 측정은 운동장 트랙을 반시계방향으로 뛰기(a_jogging_CCW) 한 결과를 보여준다. g-force 신호는 가속도 센서로 부터 지구 중력의 힘을 측정하게 되므로, 지표면

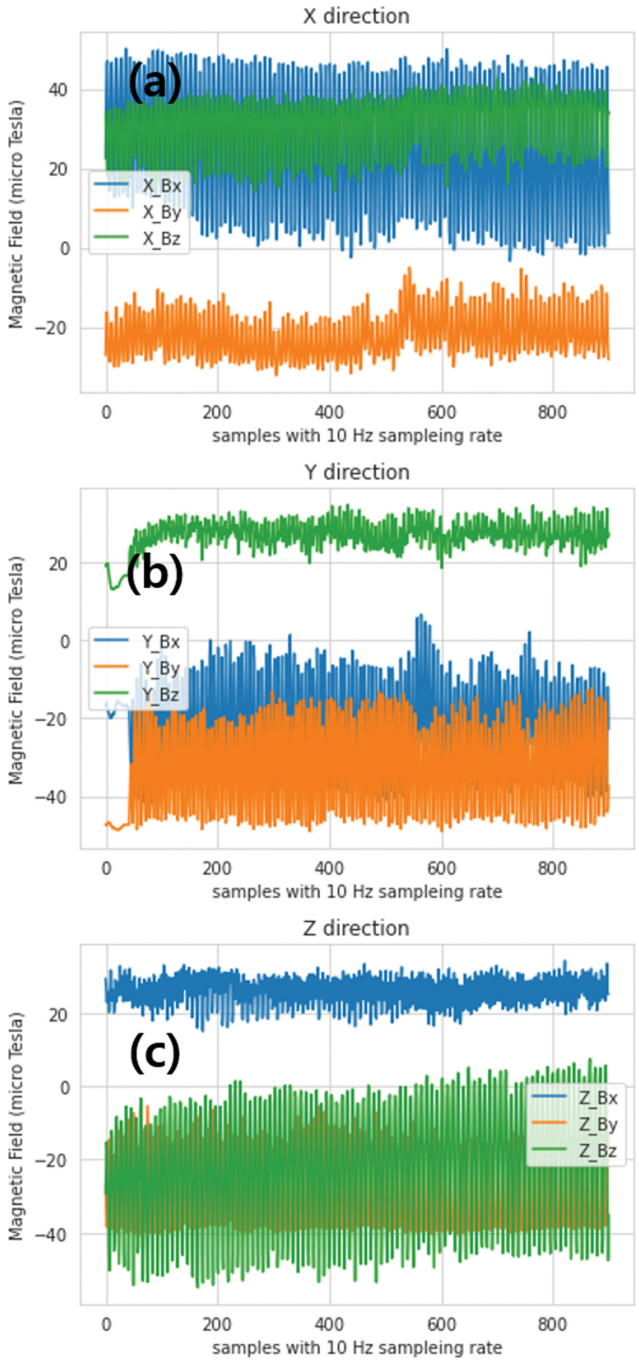


Fig. 2. (Color online) 3-axis sensor data by hand movement in a fixed posture for (a) X-motion, (b) Y-motion, and (c) Z-motion.

운동에서 자기센서처럼 방향에 의존하는 신호는 나타나지 않는다[13]. 자기센서의 데이터는 지표면에서 방향에 의존하게 나타나며, 동적인 움직임인 걷기와 뛰기의 경우에 대략 1000 샘플링(100초)마다 데이터 값이 크게 변하는 것을 Fig. 3(c), (d)를 통해서 알 수 있다. 이것은 실험자가 Fig. 3(a)에서 보는것처럼 원형의 운동장 트랙을 걷거나 뛰는 상황이기 때문이다. 방향에 의존하는 자기센서의 특성은 Fig. 5에서 다시

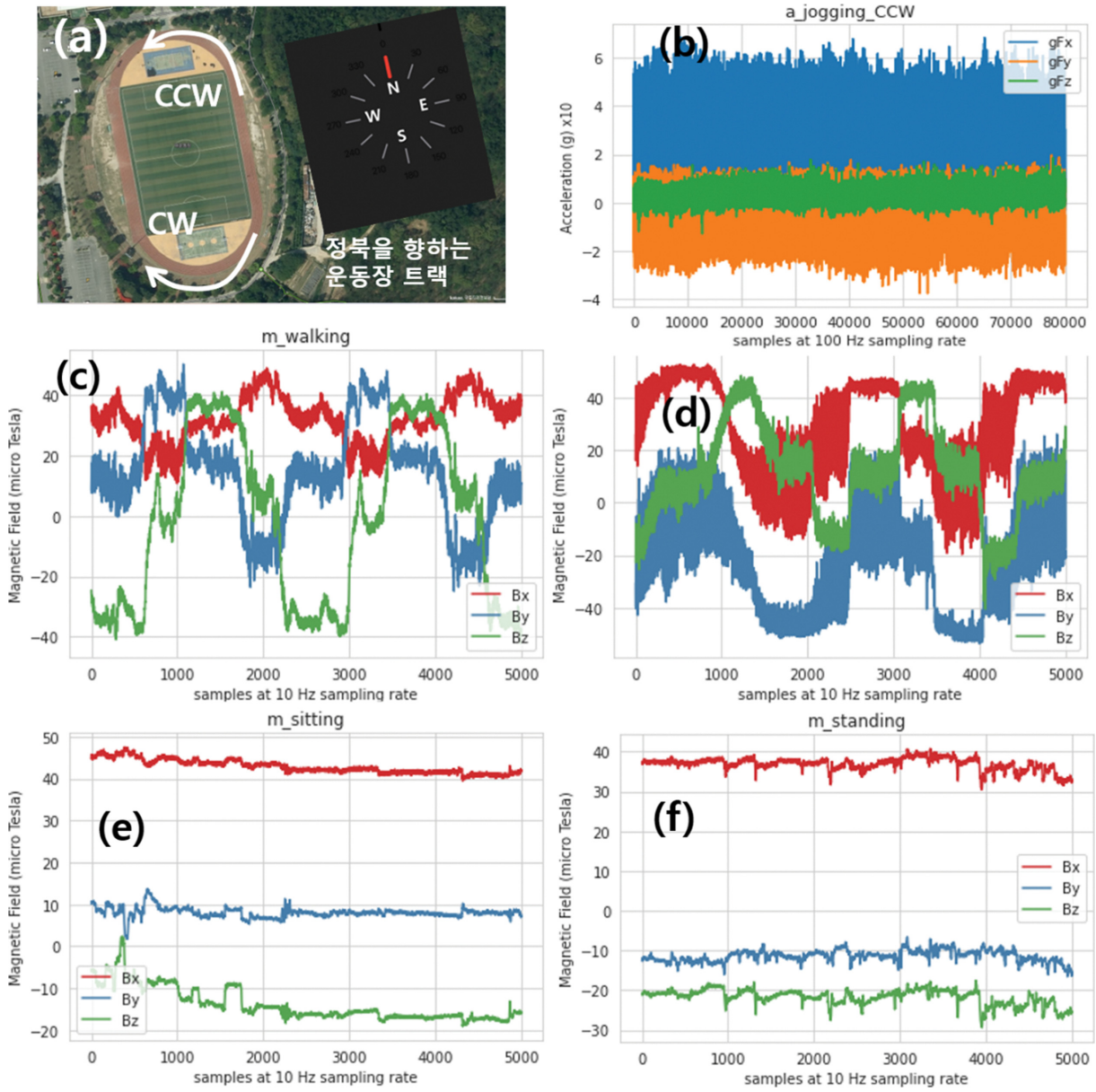


Fig. 3. (a) (Color online) Playground where data collection took place. (b) g-force measurement data for the jogging case. Four different human activities and corresponding magnetometer data for (c) the walking, (d) the jogging, (e) the sitting, and (f) the standing cases.

설명하는것으로 한다. Fig. 3(e), (f)에서 보는 것처럼 서기와 앉기의 경우에는 방향변화 없이 고정된 자세로 측정하였으므로 긴 주기의 신호변화는 없으며 짧은 주기의 불규칙적인 신호 변화가 나타난다. 이것은 측정과정에서 임의로 나타나는 손의 흔들림이나 몸의 흔들림으로 인한 신호변화로 이해할 수 있다.

Fig. 4는 운동장 트랙에서 실시한 Fig. 3의 4가지 행동인식 실험결과에 대한 통계적 분포 밀도(density)를 seaborn 모듈을 이용하여 구한 결과이다[14]. Fig. 4(a)~(c)는 Bx By, Bz 각각의 신호에 대한 4가지 행동인식에 대한 신호크기(Bx: –40 μT ~ +60 μT, By: –60 μT ~ +60 μT, Bz: –60 μT ~ +60 μT)에 대한 신호값을 행동인식에 구분에 해당하는 센서 신호값의 분포 밀도 함수를 보여준다[15]. Fig. 4에서 보이는 바와 같이 앉기 및 서기 행동(static action)에 대한 밀도 함수의 분산 값은 매우 작아서 걷기 및 달리기(dynamic action)와 구분이 잘되는 반면, 걷기와 달리기는 밀도 함수가 많이 겹치는 것을 확인할 수 있다. 특히, 달리기의 경우 개인의 달리기 속도와 불규칙적인 팔의 움직임 등으로 인하여 큰 분산 값을 갖는 것을 알 수 있다[15].

Fig. 4의 Bx, By, Bz 모든 경우에 대해서 동적 움직임인

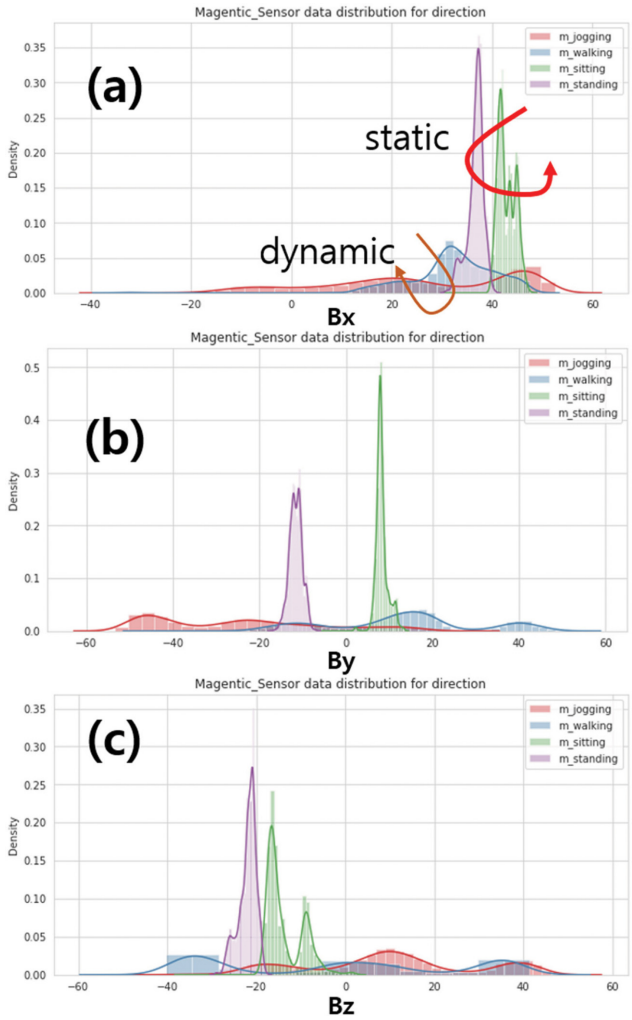


Fig. 4. (Color online) Distribution density plots for (a) Bx, (b) By, and (c) Bz during human activities.

걷기와 뛰기의 경우에는 Δ 100 μT의 신호 범위에 넓게 분포하고 있으며, 정적인 움직임인 서기와 앉기 경우에는 Δ 20 μT의 좁은 영역에 집중되어 분포하고 있음을 알 수 있다. Fig. 4에서 4가지 행동인식의 신호가 분포하는 영역은 다르지만, 분포의 범위 측면에서는 정적 움직임과 동적 움직임이 구별되는 것을 알 수 있다. 이러한 차이는 손에 잡고 있는 스마트폰의 움직임이 활발한것과 그렇지 않은 경우에 따라 구별되는 것으로 이해할 수 있다[15].

Fig. 5(a)는 반시계방향으로 운동장 트랙을 2바퀴 달릴 때(jogging)의 3축 자기센서 신호(Bx, By, Bz)의 변화이다. Fig. 5(a)의 Bx, By, Bz를 보면 2바퀴 운동에 해당하므로, 대략적인 2주기의 신호를 나타낸다. 방향에 의존하는 자기센서 신호를 확인하기 위해서 운동장 트랙 2바퀴를 시계방향(m_jogging_CW), 반시계방향(m_jogging_CCW)으로 '뛰기' 모드로 측정하여 Bx, By, Bz 각각을 비교한 것은 Fig. 5(b),

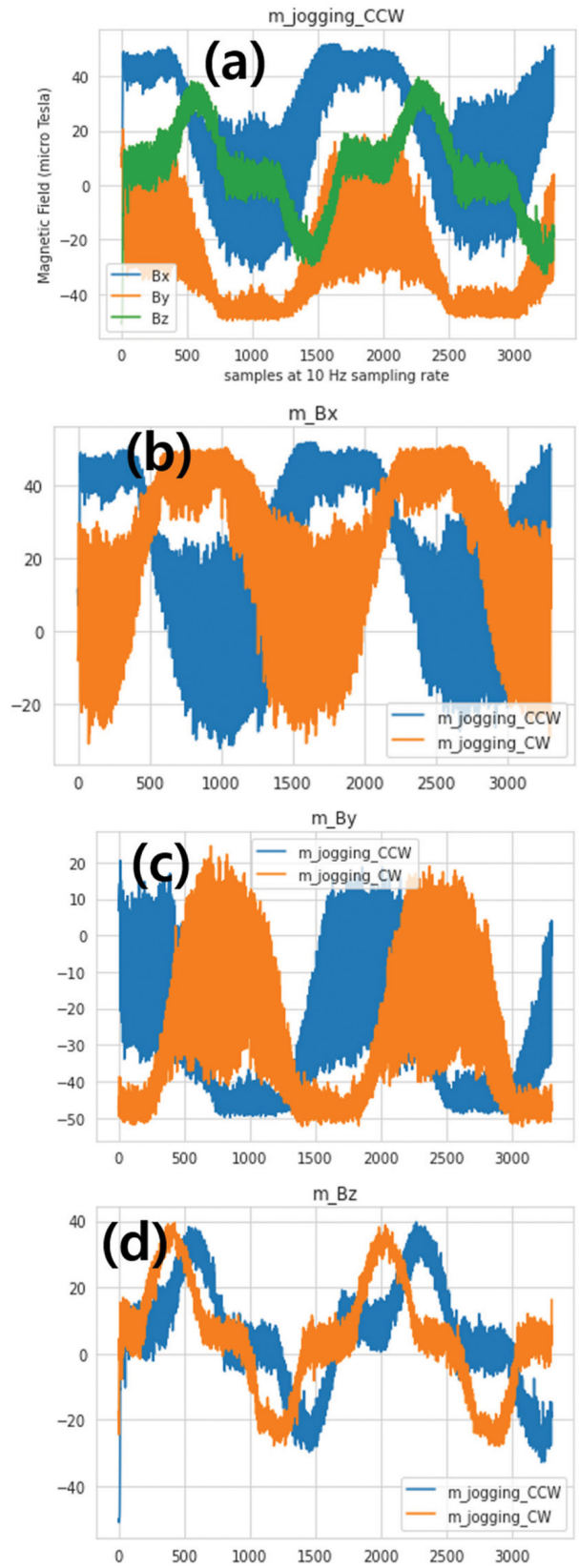


Fig. 5. (Color online) (a) Sensor data in counter-clockwise jogging. Sensor data in clockwise and counterclockwise jogging for (b) Bx, (c) By, and (d) Bz.

(c), (d)에 나타나 있다. 2바퀴 운동이기 때문에 큰 파형의 2주기를 모두 확인할 수 있으며 시계방향, 반시계 방향에 따라서 Bx, By는 신호가 역전됨을 Fig. 5(b)와 (c)를 통해서 알 수 있다. Bx와 By에서 신호 역전이 같은 샘플링에서 나타나지 않는 이유는 뛰는 주기가 정확히 일치하지 않아서 나타나는 시간 차이이며 이것은 Fig. 5(d)의 Bz 신호를 통해서 확인할 수 있다. Fig. 5(d)의 Bz 신호는 시간차이만 날 뿐, 방향에 따라서는 차이가 없음을 알 수 있는데, 이것은 Bz 신호는 수직을 향하고 있기 때문에 운동장 트랙을 뛰는 방향에는 의존하지 않는것을 의미한다.

합성곱 인공신경망(convolutional Neural Network: CNN)을 적용하기 위해서 스마트폰을 이용하여 측정하는 시작 순간과 끝나는 순간에 스마트폰 조작으로 나타나는 이상신호를 제거하기 위해서, 측정 중의 5000 샘플링 데이터(500초 동안의 연속적인 데이터)를 총 데이타로 하였다. 20 샘플링 신호(2초)를 한 프레임으로 지정하고 프레임 사이의 뜀은(hopping)은 10 샘플링(1초)으로 하여 신호의 연속적인 특성이 유지되도록 하였다. '뛰기(jogging)' 모드로 예를 들어 설명하면 Fig. 6(a)에서 보이는 것처럼 5000개 샘플 중에서 400개로 축소된 Fig. 6(b)를 살펴보자. 초당 수집되는 샘플링 수, Sampling rate은 10 Hz 이므로 2초에 해당하는 20개 샘플을 한 개의 프레임(파이썬 코딩시 변수설정: frame_size)으로 설정한다. 프레임간의 신호 손실을 줄이기 위해서 프레임 간격(파이썬 코딩시 변수 설정: hop_size)은 10개 샘플로 설정하여 전체 400개 샘플에 대해서 프레임을 약 40개 만들 수 있다. Fig. 6(c)는 이와 관련된 파이썬 코딩을 보여준다. 최종 CNN 모델에 입력되는 데이터 값의 비대칭 효과를 줄이기 위해서 sklearn 툴킷에서 제공하는 'StandardScaler' 함수를 사용하여 표준정규분포를 가지는 데이터가 입력되도록 하였다.

Fig. 7은 CNN 모델링의 요약을 보여주며, CNN을 이용한 딥러닝 방법은 convolution layer, pooling layer, fully connected layer, 그리고 최종 분류를 위한 softmax layer로 구성된다[16]. Fig. 7에서 보는 것처럼 각 프레임은 CNN에 입력되는 input으로 이해될 수 있으며 input 데이타의 형태(shape)는 (20, 3, 1)로 하여 2D CNN을 적용하였다. 여기서 20은 샘플 수, 3은 3축 자기장 센서 데이터, 1은 1 채널수를

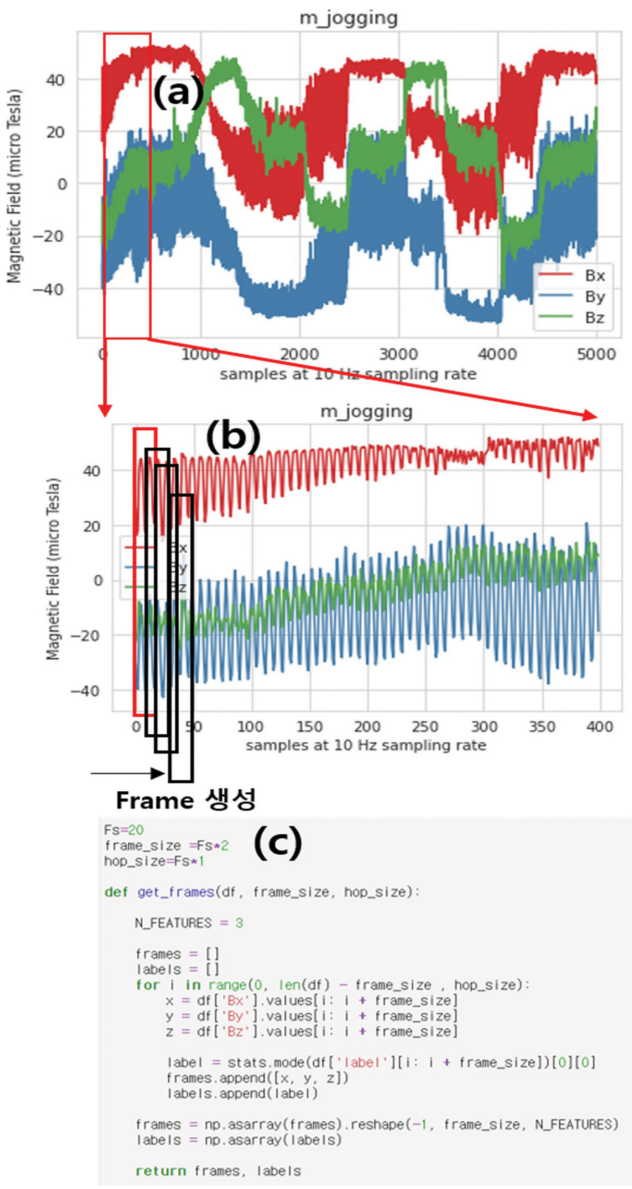


Fig. 6. (Color online) (a, b) Frame setting method in 3-axis magnetometer sensor data for CNN input and (c) python code.

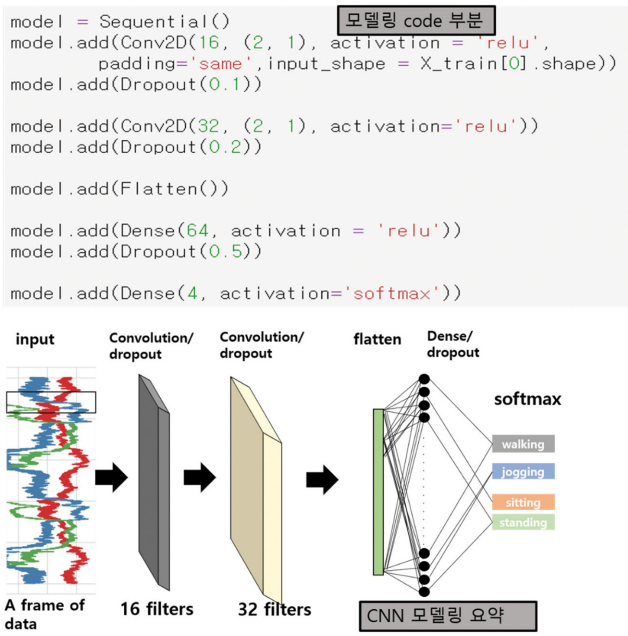


Fig. 7. (Color online) CNN modeling coding and architecture of a CNN model for sensor-based human activity recognition.

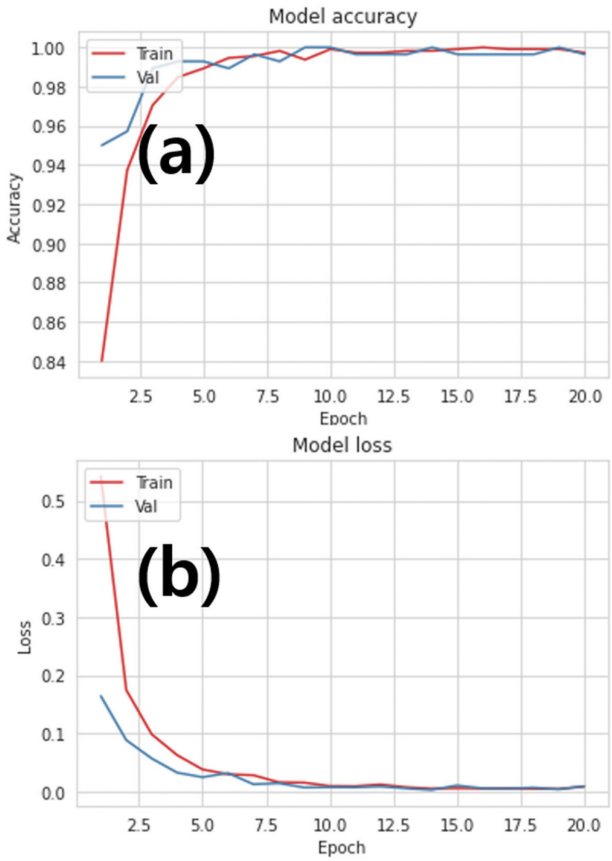


Fig. 8. (Color online) Experimental results: (a) epoch vs. accuracy and (b) epoch vs. loss.

의미한다. 학습 데이타(training data)는 1398이며, 테스트 데이타(test data)는 600으로 나누어진다. 학습과정에서 검증데이타(validation data)는 학습데이타의 20%로 설정하였다. 비용함수(loss fuction)는 분류에서 적용되는sparse_categorical_crossentropy를 사용하였으며 학습과정의 평가지표(metrics)는 정확도(accuracy)로 하였다. 학습된 CNN 모델의 정확도(accuracy)를 최종 테스트 데이타로 확인하고, scikit-learn 툴킷에서 제공하는 혼돈행렬(confusion matrix)를 통해서 각각의 행동인식에 대한 결과를 확인하였다. 모델에 적용된 주요 code 부분과 CNN 모델링 요약은 Fig. 7에 나타나 있다.

Fig. 8은 20 epoch에 대해서 전체 학습 데이타와 검증 데이타의 정확도(Fig. 8(a))와 손실함수(Fig. 8(b)) 값을 보여준다. 8 epoch 이상에서 정확도와 손실함수는 변화가 미미하며, 20 epoch로 증가하여도 과적합(overfitting)은 발생하지 않음을 알 수 있다. Fig. 9는 학습된 모델을 이용하여 테스트 데이터를 예측한 결과에 대한 실제값(true label)과 예측값(predicted label)에 대한 혼돈행렬을 보여준다. 테스트 데이타 600개는 4가지 행동인식에 대해서 150개로 골고루 분포하고 있으며, '걷기' 행동에 대해서 2개의 프레임이 '서기'로 틀린

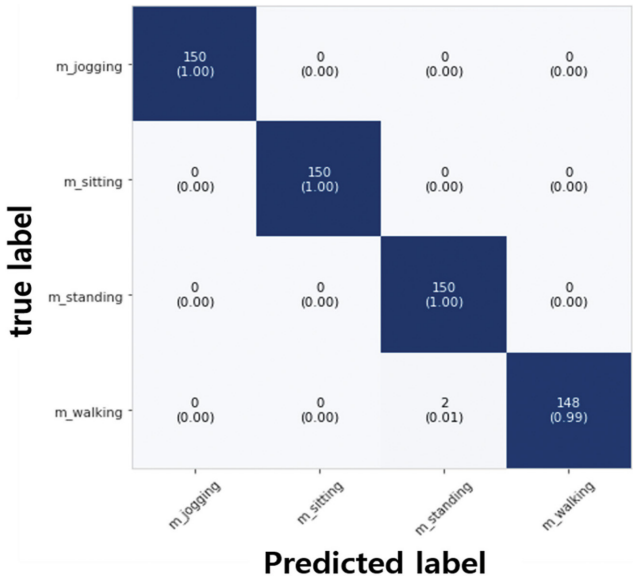


Fig. 9. (Color online) CNN model confusion matrix of 4-class human activity recognition results. The parentheses indicate the ratio.

예측 결과를 보여주며 나머지 모든 데이타는 정확히 예측함을 보여준다.

IV. 결　　론

결론으로, 모바일 스마트폰에 내장된 자기센서를 이용하여 4가지 행동인식에 대한 머신러닝 방법을 적용하여 결과를 확인하였다. 시간 함수에 대한 3축 자기센서를 2초 동안의 20 샘플링을 기준으로 프레임으로 나누어 딥러닝 방법의 하나인 CNN 방법을 적용하였다. 가속도 센서에서 많이 연구된 방법을 자기센서에 적용하여 유사한 결과를 확인 하였으며, 추후 여러 사용자에 의해서 축적된 많은 데이타에 대해서 자기센서기반 머신러닝 학습을 해볼 필요가 있다. 또한 방향에 민감한 운동에 대한 분류 작업에서 가속도 센서와 자기센서에 대한 비교를 통해서 자기센서의 장점이 드러나는지 추후 연구가 필요함을 알 수 있었다.

감사의 글

본 연구는 과학기술정보통신부 재원으로 한국연구재단의 지원을 받아 수행된 연구이며, 이에 감사드립니다(NRF-2021R1F1A1052971).

References

[1] A. Esteva, K. Chou, S. Yeung, N. Naik, A. Madani, A. Mot-

taghi, Y. Liu, E. Topol, J. Dean, and R. Socher, npj Digit. Med. **4**, 5 (2021).
[2] H. Chen and Z. Zhang, Sci. Rep. **11**, 7648 (2021).
[3] T. Zebin, P. J. Scully, and K. B. Ozanyan, 2016 IEEE SENSORS Conference, 1 (2016).
[4] H. Li and M. Trocan, Microelectron. J. **88**, 164 (2019).
[5] C. G. Pendão, A. C. Moreira, and H. Rodrigues, 2014 7th IFIP Wireless and Mobile Networking Conference (WMNC), 1 (2014).
[6] S. Saha, S. Chatterjee, A. K. Gupta, I. Bhattacharya, and T. Mondal, 2015 International Conference on Computing and Network Communications (CoCoNet), 457 (2015).
[7] R.-A.Voicu, C. Dobre, L. Bajenaru, and R.-I. Ciobanu, Sensors **19**, 458 (2019).
[8] D. Son, J. Korean Magn. Soc. **24**, 22 (2014).
[9] D. Y. Kim and S. S.Yoon, J. Korean Magn. Soc. **30**, 205 (2020).
[10] N. Golestani and M. Moghaddam, Nat. Commun. **11**, 1551 (2020).
[11] T. H. Jeon and J. K. Lee, J. Korean Soc. Precis. Eng. **36**, 659 (2019).
[12] S. Odenwald, Citizen Science: Theory and Practice **4**, 18 (2019).
[13] https://www.vieyrasoftware.net/sensors-sensor-modes.
[14] https://seaborn.pydata.org/.
[15] S.-M. Lee, H.-R. Jo, and S.-M. Yun, Inf. Commun. Mag. **33**, 65 (2016).
[16] https://www.tensorflow.org/tutorials/images/cnn.